



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

#2

In re Patent Application of:

Takeshi ISHIDA, et al.

Application No.:

Group Art Unit:

Filed: June 29, 2001

Examiner:

For: SERVICE MANAGING APPARATUS

**SUBMISSION OF CERTIFIED COPY OF PRIOR FOREIGN
APPLICATION IN ACCORDANCE
WITH THE REQUIREMENTS OF 37 C.F.R. §1.55**

Assistant Commissioner for Patents
Washington, D.C. 20231

Sir:

In accordance with the provisions of 37 C.F.R. §1.55, the applicant(s) submit(s) herewith
a certified copy of the following foreign application:

Japanese Patent Application No. 2001-046516

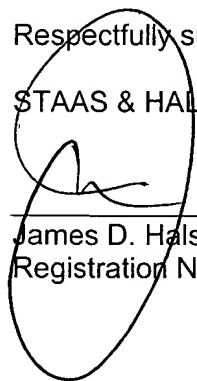
Filed: February 22, 2001

It is respectfully requested that the applicant(s) be given the benefit of the foreign filing
date(s) as evidenced by the certified papers attached hereto, in accordance with the
requirements of 35 U.S.C. §119.

Respectfully submitted,

STAAS & HALSEY LLP

Date: June 29, 2001

By: 
James D. Halsey, Jr.
Registration No. 22,729

700 11th Street, N.W., Ste. 500
Washington, D.C. 20001
(202) 434-1500

JC971 U.S. PRO
09/897100
07/03/01

PATENT OFFICE
JAPANESE GOVERNMENT

This is to certify that the annexed is a true copy of the
following application as filed with this office.

Date of Application: February 22, 2001

Application Number: Patent Application
No. 2001-046516

Applicant(s): FUJITSU LIMITED

May 31, 2001

Commissioner,

Patent Office Kozo Oikawa

Certificate No. 2001-3049187

日 本 国 特 許 庁
JAPAN PATENT OFFICE

JC971 U.S. PTO
09/897100
07/03/01

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日
Date of Application:

2001年 2月22日

出 願 番 号
Application Number:

特願2001-046516

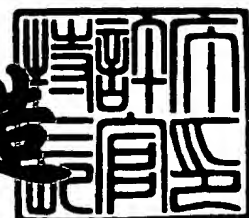
出 願 人
Applicant(s):

富士通株式会社

2001年 5月31日

特 許 庁 長 官
Commissioner,
Japan Patent Office

及 川 耕 造



出証番号 出証特2001-3049187

【書類名】 特許願

【整理番号】 0150045

【提出日】 平成13年 2月22日

【あて先】 特許庁長官殿

【国際特許分類】 H04L 12/16

【発明の名称】 サービス管理装置

【請求項の数】 5

【発明者】

 【住所又は居所】 東京都文京区後楽1丁目7番27号 株式会社富士通ビ
 ジネスシステム内

 【氏名】 石田 武

【発明者】

 【住所又は居所】 東京都文京区後楽1丁目7番27号 株式会社富士通ビ
 ジネスシステム内

 【氏名】 尹 京海

【発明者】

 【住所又は居所】 東京都文京区後楽1丁目7番27号 株式会社富士通ビ
 ジネスシステム内

 【氏名】 山本 実

【特許出願人】

 【識別番号】 000005223

 【氏名又は名称】 富士通株式会社

【代理人】

 【識別番号】 100074099

 【住所又は居所】 東京都千代田区二番町8番地20 二番町ビル3F

 【弁理士】

 【氏名又は名称】 大菅 義之

 【電話番号】 03-3238-0031

【選任した代理人】

【識別番号】 100067987

【住所又は居所】 神奈川県横浜市鶴見区北寺尾 7 - 2 5 - 2 8 - 5 0 3

【弁理士】

【氏名又は名称】 久木元 彰

【電話番号】 045-573-3683

【手数料の表示】

【予納台帳番号】 012542

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9705047

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 サービス管理装置

【特許請求の範囲】

【請求項 1】 情報装置に、ネットワークを介してクライアントからのサービス要求に応じたサービスを提供するサービスサーバを複数收容し、該複数のサービスサーバにサービス要求を配分するサービス管理方法を実現させるプログラムにおいて、

該複数のサービスサーバを、提供するサービスの品質レベル毎の複数のグループのサービスサーバと、該グループ間を移行して、移行先のグループのサービス品質でサービスを提供する中間サーバグループのサービスサーバとにグループ化して管理する管理ステップと、

いずれかのグループのサービスサーバの負荷が増加し、そのグループが提供すべき品質レベルを維持できなくなる場合に、該中間サーバグループの最も負荷が低いサービスサーバを少なくとも 1 つ、該グループのサービスサーバとして使用して、該グループのサービスサーバの負荷の低減を図る中間サーバ移行ステップと、

を備えるサービス管理方法を情報装置に実現させることを特徴とするプログラム

。

【請求項 2】 前記管理ステップは、

前記グループ化されたサービスサーバが、どのグループに属するかの情報を格納する格納手段を更に備えることを特徴とする請求項 1 に記載のプログラム。

【請求項 3】 前記サービスの品質は、前記サービスサーバの応答時間であることを特徴とする請求項 1 に記載のプログラム。

【請求項 4】 前記サービス要求の履歴を記録するログ管理ステップと、

該ログ管理ステップの記録に基づいて、日にちあるいは曜日毎にスケジュールを作成し、作成したスケジュールに従って自動で前記グループ分けの仕方を変更させるスケジュール管理ステップと、

を更に備えることを特徴とする請求項 1 に記載のプログラム。

【請求項 5】 前記各サービスサーバは、自サーバがサービス要求を処理する

ために必要とする負荷値を計測する負荷計測ステップを有し、

該負荷計測手段から報告される各サービスサーバの負荷値に基づいて、前記中間サーバグループのサービスサーバを別のグループに移行させることを特徴とする請求項 1 に記載のプログラム。

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】

本発明は、サービスを提供するサービスサーバにサービス要求を分配するサービス管理装置に関する。

【 0 0 0 2 】

【従来の技術】

今日、インターネットの普及により、インターネット上での様々なビジネスが展開されつつある。中でも、インターネットを介して、ユーザに様々なアプリケーションサービスを提供する A S P (Application Service Provider) サービスが実用化されている。

【 0 0 0 3 】

図 1 7 は、A S P サービスを提供するシステムの概略構成図である。

サービスを行う複数のサービスサーバ 1 0 は、サービス要求のサービスサーバ 1 0 への分配などの管理を行うサービス管理サーバ 1 1 に接続され、サービス管理サーバ 1 1 を介して、クライアント 1 4 からのサービス要求を受け付ける。また、サービス管理サーバ 1 1 は、ウェブサーバ 1 2 に接続され、クライアント 1 4 からのサービス要求をインターネット 1 5 経由で受け付けるよう構成される。

【 0 0 0 4 】

このような、ウェブサーバ 1 2、サービス管理サーバ 1 1、及びサービスサーバ 1 0 からなるシステムは、データセンタ 1 3 と呼ばれ、様々なプロバイダからのアプリケーションの提供サービスをまとめて管理する。

【 0 0 0 5 】

ところで、最近では、アプリケーションの提供サービスにおいて、受け付けたサービス要求を単にサービスサーバ 1 0 に割り振って、サービスを提供するだけ

でなく、サービスの提供品質を契約によって補償しつつ、アプリケーション提供サービスをクライアント 14 に提供するという S L A (Service Level Agreement : サービスの品質を保証する契約) を実現しようとしている。

【0006】

図 18 は、S L A におけるサービス管理サーバのサービス管理方法の従来技術を説明する図である。

S L A において補償されるサービスの品質としては、サービスサーバの応答速度、障害復旧時間、障害発生時の補償などが挙げられるが、以下の説明においては、サービスの品質としてサービスサーバの応答速度を念頭に置いて説明する。

【0007】

従来の S L A の実現方法の一つとしては、図 18 (a) に示されるような、提供するサービスの品質毎にサービスサーバをグループ分けする方法がある。この場合、サービスサーバは、高品質のサービスを提供するサーバと、一般レベルのサービスを提供するサーバなどのグループに分けられ、それぞれのグループ内では、定められたサービスの品質を維持しつつサービスの提供を行うように構成される。サービス管理サーバは、サービス要求の品質契約内容に従って、受信したサービス要求をどのグループのサービスサーバに割り振るかを決定して、サービス要求 (リクエスト) を送信する。

【0008】

このような場合、それぞれのグループで品質を統一して維持することは容易であるが、リクエストが一部のレベルに集中した場合、サーバ間の負荷の格差が大きくなり、全てのサーバ資源を有効に活用できないと言う問題が生じる。すなわち、サービス品質がサービスサーバの応答速度である場合、高品質のサービスを提供するサービスサーバは、クライアントからのサービス要求に対して高速に応答する必要があるので、高品質サービスサーバには、大きな負荷がかからないようにサービス要求を割り振る必要がある。このようにすれば、高品質サービスサーバは、常時高品質のサービスを提供することができるので、サービス品質の管理は行いやすい。しかし、上記したように、サービス品質レベル毎にサービスサーバがグループ分けされているので、一部のグループにサービス要求が集中して

も、サービス要求の処理はグループ内で行わなければならない、グループ間での負荷の格差、すなわち、サーバ資源の無駄が生じる。

【 0 0 0 9 】

また、従来の S L A の実現方法の他の方法としては、図 1 8 (b) に記載されているように、全てのサービスサーバが全ての品質レベルのサービス进行处理するという方法がある。この場合、各サービスサーバは、高品質契約のサービス要求（リクエスト）を優先的に処理すると共に、一般レベル契約のリクエストも受け、処理しなければならない。このとき、サービス管理サーバは、リクエストの品質契約内容を知る必要はなく、各サービスサーバの負荷が出来るだけ均等になるように、各リクエストを各サービスサーバに割り振る処理を行う。

【 0 0 1 0 】

従って、各サービスサーバは、リクエストが高品質契約のものか、一般レベルのものかを判断し、判断結果によって処理の優先度を変えて処理するという手続きを行わなければならない。これを実現するためには、サービスサーバに複雑なロジックからなるプログラムを実装しなければならない。

【 0 0 1 1 】

このような場合、全てのサーバ資源を均等に使用でき、負荷分散は行いやすいが、ロジックが複雑になる上に、品質を正確に維持することが困難になるという問題点がある。また、運用中の A S P サービスにこの方式を導入しようとする、個々のアプリケーションサービスを根本的に作り直す必要があるため、時間的にも金銭的にも莫大なコストが必要になる。

【 0 0 1 2 】

【発明が解決しようとする課題】

図 1 9 は、従来の問題点を説明する図である。

上記したように、S L A において、確実にサービス品質を確保しようとするので有れば、図 1 8 (b) の構成は採用できず、図 1 8 (a) の構成を採用する必要があるが、前述したように、グループ間で負荷の格差が生じ、サーバ資源が有効に使用できないという問題が生じる。

【 0 0 1 3 】

本発明の課題は、サービスの品質を維持しつつ、サーバの負荷を適切に分散することの出来るサービス管理装置を提供することである。

【 0 0 1 4 】

【課題を解決するための手段】

本発明のサービス管理装置は、ネットワークを介してクライアントからのサービス要求に応じたサービスを提供するサービスサーバを複数收容し、該複数のサービスサーバにサービス要求を配分するサービス管理装置において、該複数のサービスサーバを、提供するサービスの品質レベル毎の複数のグループのサービスサーバと、該グループ間を移行して、移行先のグループのサービス品質でサービスを提供する中間サーバグループのサービスサーバとにグループ化して管理する管理手段と、いずれかのグループのサービスサーバの負荷が増加し、そのグループが提供すべき品質レベルを維持できなくなる場合に、該中間サーバグループの最も負荷の低いサービスサーバを少なくとも1つ、該グループのサービスサーバとして使用して、該グループのサービスサーバの負荷の低減を図る中間サーバ移行手段とを備えることを特徴とする。

【 0 0 1 5 】

本発明によれば、サービスサーバをサービスの品質レベル毎にグループ化してサービスを提供するので、安定した品質でサービスを提供できる。また、その場合に問題となるグループ間での負荷の偏りを、グループ間を動的に移行可能な中間サーバグループのサービスサーバを設け、これを負荷の多いグループへ移行させることにより、解消するので、安定した品質でのサービスの提供と、サービスサーバ間の負荷を適切に分散することが出来る。

【 0 0 1 6 】

【発明の実施の形態】

図1は、本発明の実施形態の概略を示す図である。

本実施形態の説明においては、クライアント（顧客）とのサービス契約において、上位レベルの品質と下位レベルの品質の契約のみがあるものとする。そして、この場合、本実施形態では、上位レベルのサービスサーバグループと下位レベルのサービスサーバグループとを用意すると共に、中間グループのサービスサー

バを用意する。上位レベルグループのサービスサーバは、上位レベル品質のリクエストを専用に受け付け、下位レベルグループのサービスサーバは、下位レベル品質のリクエストを専用に受け付ける。中間グループのサービスサーバは、通常時は、上位レベルの品質で、下位レベルのリクエストを処理する。すなわち、下位レベルのリクエストを上位レベルの品質でサービス提供していることになる。

【 0 0 1 7 】

そして、例えば、上位レベルグループのサービスサーバにリクエストが集中し、負荷が大きくなり、品質の維持が難しくなるとすると、中間グループのサービスサーバをレベルアップさせ、下位レベルのリクエストではなく、上位レベルのリクエストを処理させるようにする。

【 0 0 1 8 】

この場合、上記のようなリクエストの割り振りは、サービス管理サーバが一括して行う。

図 2 は、本発明の実施形態が適用されるシステムの構成図である。

【 0 0 1 9 】

クライアント（不図示）は、インターネット 2 0 を介して、ウェブサーバ 2 2 にリクエストを送信する。ウェブサーバ 2 2、サービス管理サーバ 2 4、サービスサーバ 2 5 - 1 ~ 2 5 - n、及びデータベースサーバ 2 6 からなるデータセンタは、ファイアウォール 2 1 によって外部からの不正なアクセスに対して防御される。真正なクライアントは、このファイアウォール 2 1 をパスする ID などのアカウントを有しており、これを使って、ウェブサーバ 2 2 にリクエストを通知する。ウェブサーバ 2 2 では、このリクエストをトリガにして起動し、後段のサービス管理サーバ 2 4 などに命令を通知するサーバレットエンジン 2 3 が設けられている。従って、ウェブサーバ 2 2 がクライアントからのリクエストを受け取ると、サーバレットエンジン 2 3 が、このリクエストに基づいた処理を行うべき旨の命令をサービス管理サーバ 2 4 に通知する。

【 0 0 2 0 】

サービス管理サーバ 2 4 は、ウェブサーバ 2 2 のサーバレットエンジン 2 3 から受け取った命令に基づいて、リクエストを適切なサービスサーバ 2 5 - 1 ~ 2

5-nに振り分ける。通常状態では、上位レベルのリクエストは、上位レベルグループのサービスサーバに、下位レベルのリクエストは、中間グループあるいは、下位レベルグループのサービスサーバに受け渡す。サービスサーバ25-1～25-nは、データベースサーバ26に格納されている情報から、サービスの提供に必要なデータを取得し、アプリケーションサービス内容に変換して、クライアントに送り返す。

【 0 0 2 1 】

ここで、図1で説明したような処理をサービス管理サーバが行うためには、上位、下位レベル、中間グループのサービスサーバを管理するための管理用データが必要となる。以下は、本実施形態において必要とする管理用データの一例である。

・ サービス管理サーバで管理されているデータ

ー サービスサーバ情報（サービスサーバ1台毎に定義されている）

ID=Server A：サーバを識別するID

Load=20：現在のサービスサーバの負荷値（サービスサーバが定期的にサービス管理サーバに通知する）

Limit __HighLV=50、Limit __LowLV=100：閾値（サービスレベル毎の品質を維持するための負荷値の限界）

Group=上位：サーバが所属しているグループ

ReqLV=上位：実行するリクエストのレベル

ResLV=上位：維持しなくてはならない品質のレベル

serviceX=5、serviceY=10：サービス処理の負荷値（そのサーバで各サービスの処理を実行した時の負荷値）

ー 共通情報

changeTime= 午前3：00：サーバ構成の自動変更を行う時間

Priority1=曜日、Priority2=日にち：リクエスト集計の優先順位（リクエストを集計する時にどの条件の偏りを優先するかを示す）

schedule：サーバ構成のスケジュールデータ

・ 各サービスサーバで管理されている情報

runningService= {X,X,X,Y} : 実行中の処理

サービスサーバ情報（サービス管理サーバで管理されているものと同じ）

・クライアントからのリクエストから取得できる情報

Service=X : 実行する処理の種類

SL= 上位 : 契約しているサービスの品質レベル

図 3 は、中間グループのサービスサーバ（中間サーバ）の状態変化を示す図である。

【 0 0 2 2 】

図 1 8 （ b ） に示したように、従来の負荷分散の手法では、サーバやネットワーク上の負荷を計測し、最も負荷の低いサーバに処理を行わせるというものが一般的であった。しかし、そのようなサービスの品質を考慮しないやり方では、SLA を守ることと負荷分散を両立することは出来ない。

【 0 0 2 3 】

本実施形態では、中間サーバのレベル変化の際に維持すべき品質を細かく管理することによって負荷分散と品質の維持を両立する。

図 3 に示されるように、中間サーバは、通常状態では、上位レベルの品質で、下位レベルのリクエストを処理している。ここで、上位レベルグループのサービスサーバの負荷が大きくなったとすると、前述したように、上位レベルにレベルアップして、上位レベルの品質で、上位レベルのリクエストを処理する。ここで、通常状態で上位レベルへの品質を維持しているため、レベルアップして上位レベルのリクエストを受け入れても中間サーバは品質を維持することが出来る。

【 0 0 2 4 】

また、前述の説明では記載しなかったが、中間サーバは、上位から通常、通常から下位、下位から通常の状態遷移もすることができる。

すなわち、中間サーバが上位レベルにある場合に、実行中の上位レベルのリクエストが無くなったら、通常状態に戻す。したがって、中間サーバが上位レベルにあって、上位レベルのリクエストを処理し終わると、下位レベルのリクエストを処理しはじめる。これにより、通常状態に戻る。もし、中間サーバが、下位レベルグループの負荷が大きくなって、下位レベルにレベルダウンする場合でも、

通常状態から下位レベルに移行するので、上位レベルのリクエストを処理した状態で、下位レベルのリクエストを処理しはじめることはない。

【 0 0 2 5 】

中間サーバが下位レベルに移行した場合には、下位レベルの品質で、下位レベルのリクエストを処理する。通常状態では、下位レベルのリクエストのみを実行しているため、レベルダウンしてサービスの品質を落としても問題は生じない。また、下位レベルグループのサービスサーバの負荷が下がって、上位レベルの品質を守ることができるようになったら、中間サーバは、通常状態に戻る。すなわち、下位レベルのリクエストを上位レベルの品質で処理するようにする。通常状態に戻った後は、前述したように、すぐに上位レベルにレベルアップすることができる。

【 0 0 2 6 】

図 4 は、サービス管理サーバの処理の概略を示す図である。

まず、ステップ S 1 において、サービス管理サーバがリクエストを受信する。次に、ステップ S 2 において、リクエストを振り分けるサービスサーバを決定する。また、このとき、中間サーバのレベルアップ（ダウン）が必要か否かの判断を行う。そして、ステップ S 3 において、振り分け先に決まったサービスサーバにリクエストを送信する。

【 0 0 2 7 】

ステップ S 2 及び S 3 の処理の詳細を以下に説明する。

図 5 は、サービス管理サーバの処理の詳細を示すフローチャートである。

図 5 において、左側に示されているフローは、図 4 と同じものであるが、ここでは、説明を簡略化するため、受信するリクエストをサービス X に対する上位レベルのリクエストとしている。

【 0 0 2 8 】

図 5 において、右側に示されているのは、左側に示されているフローのステップ S 2、S 3 の詳細フローである。

まず、ステップ S 1 において、サービス X に対する上位レベルのリクエストをサービス管理サーバが受け取ると、ステップ S 1 3 において、上位レベルグルー

プのサービスサーバでサービス X を実行可能なサーバが存在するか否かを調べる。ステップ S 1 3 の詳細については、図 6 で説明する。ステップ S 1 3 において、実行可能なサーバが存在すると判断された場合には、ステップ S 1 7 に進んで、実行可能なサーバの内、負荷の最も低いサーバにリクエストを割り振る。

【 0 0 2 9 】

ステップ S 1 3 において、実行可能なサーバが無いと判断された場合には、ステップ S 1 4 において、レベルアップした中間サーバがもしあれば、その中でサービス X を実行可能なサーバが存在するか否かを調べる。ステップ S 1 4 の詳細についても図 6 で説明する。ステップ S 1 4 において、実行可能なサーバが存在すると判断された場合には、ステップ S 1 8 において、実行可能なサーバの内、負荷の最も低いサーバにリクエストを割り振る。

【 0 0 3 0 】

ステップ S 1 4 において、実行可能なサーバが存在しないと判断された場合には、ステップ S 1 5 に進んで、通常状態の中間サーバが存在するか否かを判断する。ステップ S 1 5 において、通常状態の中間サーバが存在すると判断された場合には、ステップ S 1 9 において、中間サーバの内、1 台を上位レベルにレベルアップさせる。ステップ S 1 9 の詳細については、図 7 で説明する。そして、ステップ S 2 0 において、レベルアップした中間サーバにリクエストを割り振る。

【 0 0 3 1 】

ステップ S 1 5 において、通常状態の中間サーバが存在しないと判断された場合には、ステップ S 1 6 において、実行不能なため、リクエストを待機管理手段に送信する。待機管理手段とは、サービスサーバが混雑している際に、リクエストを待機させるサーバのことである。

【 0 0 3 2 】

図 6 は、図 5 のステップ S 1 3、S 1 4 の詳細を示すフローチャートである。

なお、図 6 においては、上位レベルグループ内のサーバに対する処理として示しているが、レベルアップした中間サーバを対象とする場合も同様である。

【 0 0 3 3 】

まず、ステップ S 2 5 において、調べるべきサーバ（サーバ A とする）のサー

バ情報を取得する。ここで取得するサーバ情報の例が図 6 右上に示されている。そして、ステップ S 2 6 において、リクエストが実行可能か否かを判断する。今の場合、図 6 の右上のサーバ情報によれば、実行不可能となる。なぜなら、現在の負荷が 5 0 であり、リクエストを実行することによって負荷が 5 増えるので、上位レベルの品質を保持するのに必要な負荷の限界が 5 0 であるので、リクエストを受け付けると、上位レベルの品質を保持できなくなってしまうからである。従って、不可能な場合には、ステップ S 2 8 に進む。可能な場合には、ステップ S 2 7 において、実行可能なサーバとして情報をストック（記憶）し、すでに実行可能なサーバを記憶していた場合には、サーバ負荷の余裕（閾値との差）が大きい方を記憶して、ステップ S 2 8 に進む。

【 0 0 3 4 】

ステップ S 2 8 においては、上位レベルの全てのサーバに対し判定を行ったか否かを判断し、N O の場合には、ステップ S 2 5 に戻り、Y E S の場合には、ステップ S 2 9 に進む。ステップ S 2 9 においては、実行可能なサーバが存在するか否かを判断し、存在する場合には、ステップ S 3 0 において、実行可能なサーバの内、最も負荷の低いサーバに割り振り、存在しない場合には、ステップ S 3 1 において、上位レベルグループには実行可能なサーバが存在しないと判断する。

【 0 0 3 5 】

図 7 は、図 5 のステップ S 1 9 を詳細に示したフローチャートである。

ここでは、中間サーバ F をレベルアップさせるものとする。

まず、ステップ S 3 5 において、サービス管理サーバのサービスサーバ情報の内、サーバ F の所属グループ情報を書き換え、それに伴って処理リクエストレベルなどの情報も書き換える。図 7 の右上に記載されている例では、実行すべきリクエストのレベル ReqLV が下位から上位に書き換えられている。また、維持すべき品質は上位のままである。次に、ステップ S 3 6 において、サービス管理サーバは、サービスサーバ F（中間サーバ）にレベルアップを通知する。そして、ステップ S 3 7 において、サービスサーバ F は、自分の有するサーバ情報を書き換える。すなわち、図 7 の右下に記載されているように、ReqLV を下位から上位に

書き換える。

【 0 0 3 6 】

次に、各サービスサーバの負荷を自動計測する実施形態について説明する。

各サービスサーバでのサービスの品質を維持していくためには、実行する各サービスの重さ（負荷値）を正しく把握しておく必要がある。しかし、サービスの実行にかかる負荷は実行するサービスサーバの能力や状態にも依存するため、静的な方法で定義した場合、正確さにかけるという問題がある。しかし、新しいサービスをインストールするたびに手動で計測を行うのは管理者の負担になるだけでなく、運用に支障を来す可能性がある。そこで、サービスの運用を行いながら自動的にサービスを実行する際の負荷を計測するようにする。

【 0 0 3 7 】

このようにサービスの運用を行いながらサービスを実行するための負荷を計測することによって、

- ・環境の変化が生じてもサービスの負荷をリアルタイムで計測、修正することによって常に正確な情報を把握することができる。その結果、的確なリクエストの分配が可能になり、S L A を守りやすくなる、
 - ・新しいサービスをインストールした場合でも負荷計測のためにサービスを停止させる必要がない、
- という利点が得られる。

【 0 0 3 8 】

図 8 は、サービス負荷の自動計測処理の流れを示す図である。

まず、ステップ S 4 0 において、サービス管理サーバからリクエストの割り振り先を決定し、リクエストをサービスサーバに送信する。サービスサーバでは、ステップ S 4 1 において、リクエストを受信し、ステップ S 4 2 において、実行中の処理はあるか否かが判断される。実行中の処理がある場合には、ステップ S 4 3 に進み、リクエストを普通に実行する。ステップ S 4 2 において、実行中の処理がないと判断された場合には、ステップ S 4 4 に進んで、リクエストを実行し、処理にかかった時間を計測する。そして、ステップ S 4 5 において、計測した時間を元に、その処理の負荷値を計算して、ステップ S 4 6 に進む。ステップ

S 4 6 においては、今回の計測した負荷値とこれまでの負荷値の間を取るなどして、新しい負荷値を算出する。そして、ステップ S 4 7 において、サービス管理サーバに新しい負荷値を通知する。サービス管理サーバでは、ステップ S 4 8 において、該当サーバのサービスの負荷情報を更新する。

【 0 0 3 9 】

図 9 は、サービス負荷の自動計測処理を具体的に示すフローチャートである。

図 9 左に示す概略フローでは、まず、サービス管理サーバからサービスサーバ A にリクエストが渡され、サービスサーバにおいて、要求されたサービス X を実行し、実行結果をサービス管理サーバを介してクライアントに返す処理からなっている。これらの内、要求されたサービス X を実行するステップにおいて、負荷計測が行われるが、これを詳細に示したのが、図 9 右のフローである。

【 0 0 4 0 】

まず、ステップ S 5 0 において、サービスサーバは、実行中の処理はあるか否かについて判断を行う。今の場合、サービスサーバは、自分の情報（サービスサーバ A の情報）の内、`runningService = {X,X,Y}` という情報を参照する。今の場合には、実行中のサービスがあるので、ステップ S 5 1 に進み、要求されたサービス X を実行する。ここで、実行中のサービスがない場合には、ステップ S 5 2 に進み、要求されたサービス X を実行し、処理にかかる時間を計測する。そして、ステップ S 5 3 において、処理にかかった時間を元に、そのサービス X の負荷値を計算する。負荷値としては、処理にかかった時間をそのまま用いても良いし、サービスを実行する際の CPU 等の占有時間などを用いても良い。

【 0 0 4 1 】

そして、ステップ S 5 4 において、計測によって得られた負荷値を元にそのサービス X の負荷値を更新する。ステップ S 5 5 では、更新されたサービス X の負荷値をサービス管理サーバに通知し、サービス管理サーバが管理しているサーバ情報の中のサービス負荷値を更新する。

【 0 0 4 2 】

ここで、ステップ S 5 4 と S 5 6 の更新の様子が図 9 右端に図示されている。すなわち、サービスサーバ A がローカルに持っているサーバ情報のサービス X の

負荷値serviceX=5をserviceX=6に（今の場合、負荷値の計測の結果、6という負荷値が得られたとする）、サービス管理サーバが持っているサービスサーバAの情報の負荷値serviceX=5もserviceX=6に変更する。

【0043】

上記実施形態の場合、各グループのサービスサーバの台数が固定されている場合には、中間サーバだけでは処理しきれないリクエストの偏りが生じる可能性がある。したがって、これを解消する必要がある。具体的には、リクエストのログを解析して、曜日や日にちに依存したリクエストの偏りを見つけだし、それを元にサービスサーバの運用スケジュールを設定して、自動的にサーバのグループ分けを行うようにする。

【0044】

図10は、サービスサーバの運用スケジュールを設定する実施形態の概念を示す図である。

ここで、中間サーバの取り扱いについては任意性があるが、以下の説明では中間サーバの台数は固定とし、上位、下位レベルのサーバのみを変更するものとして説明を行う。

【0045】

例えば、過去のリクエスト数の比率から、月曜日には上位レベルのリクエスト数の比率が大きく、火曜日には、上位レベルと下位レベルの比率が同じ程度であると判断されたとすると、月曜日には、上位レベルのサービスサーバの台数を多くし、下位レベルのサービスサーバの台数を少なく設定する。また、火曜日には、上位レベルと下位レベルのサービスサーバの数を同程度に設定する。しかしながら、上位、下位レベルのサーバ台数を決定する際には中間サーバの存在も考慮しなくてはならず、従って、火曜日にはリクエスト数が上位と下位とで同数だとしても上位と下位のサーバ台数を同数にすれば良いと言うわけではない。

【0046】

前述の実施形態の場合、サービスサーバのグループ分けは静的な方法のみで行われており、管理者が手動で調整するしかなかったが、その結果リクエストの偏りがそのまま負荷の偏りを生じさせ、パフォーマンスの低下を招く可能性がある

。しかし、上記実施形態によれば、過去のリクエスト数を元にサーバ構成のスケジュールを立てるので、曜日や特定日に依存したリクエストの偏りに対応でき、リクエスト数に最適化されたサービスサーバ数を各グループに配することによって、中間サーバのレベルアップ、ダウン回数が減り、サービス管理サーバの負荷軽減につながる。また、スケジュールに従って自動的にサービスサーバの構成を変更させるので、管理者の負担や人為的なミスを減らすだけでなく、サーバの構成変更にかかる時間も短縮化できるという利点がある。

【 0 0 4 7 】

図 1 1 は、スケジュールの設定処理の概略フローを示す図である。

まず、ステップ S 6 0 において、リクエストのログを取得する。そして、ステップ S 6 1 において、リクエストの比率を解析して、サービスサーバ台数の構成のスケジュールを立てる。ここで、リクエスト比率からサーバの台数構成を算出するのは運用環境や設定の細かさ（リクエスト毎に行うのか、単にレベルのみで判断するかなど）によって計算方法が変わるが、基本的にはリクエスト数が多いレベルに多くのサーバを割り当てる用にする。詳細については、本発明を利用する当業者によって適宜決定されるべきことである。

【 0 0 4 8 】

ステップ S 6 2 においては、システムの立てたスケジュールを管理者が修正し、ステップ S 6 3 において、スケジュールを設定する。ステップ S 6 2 の管理者がスケジュールを修正する処理は任意であり、スケジュールの作成から運用まで、全て自動で行うこともできるが、作成されたスケジュールを管理者が修正することも可能であるという意味である。

【 0 0 4 9 】

図 1 2 は、スケジュール設定処理のより具体的なフローチャートである。

まず、ステップ S 7 0 において、スケジュール作成システムの起動を行う。そして、ステップ S 7 1 において、サービス管理サーバのログ管理手段で記録したリクエストの処理経過である、リクエストログを取得する。リクエストログとは、図 1 2 右の（１）に示されているような記録である。そして、ステップ S 7 2 において、ログを優先順位にしたがって解析し、サーバ構成のスケジュールを作

成する。すなわち、図12右の(1)のログ管理手段の情報と(2)のサービス管理サーバの情報とを参照し、図12右の(3)に示すようなスケジュールを作成する。次に、必要であれば、ステップS73に進んで、管理者が作成されたスケジュールを修正し、ステップS74において、スケジュールを設定する。すなわち、図12右の(3)のスケジュールをサービス管理サーバに保存する。

【0050】

図13は、スケジュールに基づいてサービスサーバの構成を変更する際の処理を示すフローチャートである。

まず、ステップS80において、定義されたタイミングでサービスサーバ構成変更システムを起動する。このとき、サービス管理サーバの情報であるchangeTime=午前3:00などの情報を参照する。今の場合、午前3時にサービスサーバの構成変更を行う旨が定義されている。次に、ステップS81において、サービス管理サーバ内の変更を行うサービスサーバの所属グループの情報を更新する。すなわち、図13右の(1)の情報を参照し、図13右の(2)のサービス管理サーバ内の該当サービスサーバの情報を更新する。そして、ステップS82において、変更を行ったサービスサーバに変更通知を行う。ステップS83においては、通知を受けたサービスサーバは、図13右の(3)のサービスサーバの情報の内、所属グループの情報を変更する。また、このとき、所属グループが行うべきリクエスト受付のレベル及び維持すべき品質のレベルの設定も所属グループの設定に合わせて変更する。そして、ステップS84において、全ての変更が終わったか否かを判断し、全ての変更が終わっていない場合にはステップS81に戻って変更処理を繰り返し、全ての変更が終わった場合には、処理を終了する。

【0051】

次に、図12のステップS72について詳細に説明する。

スケジュールの作成に際しては、基本的にサービスサーバの台数は集計したリクエストの比率の従ってサービスサーバを各レベルに配分して決定する。

【0052】

例えば、サービスサーバが全部で7台あり、うち中間サーバが2台あるとする。ログを集計した結果、ある曜日の平均リクエスト数が

上位レベル：200リクエスト

下位レベル：100リクエスト

だったとする。

【0053】

リクエスト数の比率は上位：下位＝2：1であるから、中間サーバを除いた5台のサービスサーバを2：1の比率で配分することによってサーバ台数が決定される。よって、この場合、各レベルのサービスサーバの台数は次のように決定される。

上位レベル： $5 \times (2/3) = 3.333 \rightarrow 3$ 台

下位レベル： $5 \times (1/3) = 1.666 \rightarrow 2$ 台

次に、中間サーバの存在を考慮する。すなわち、中間サーバは通常下位レベルのリクエストを実行しているため、中間サーバのことを考慮しないと下位レベルにサーバを多く割り当て過ぎることになる。

【0054】

例えば、上記の例では、上位レベルに3台、下位レベルに2台のサービスサーバを割り当てたが、中間サーバの台数が2台だったので、通常時上位レベルのリクエストを実行するのが3台だけなのに対して、下位レベルのリクエストを実行するのは下位レベルのサーバと中間サーバの合わせて4台になってしまう。従って、中間サーバは下位レベルに含まれるものとして考える必要がある。

【0055】

例えば、上記例では、リクエスト数の比率は上位：下位＝2：1であったので、中間サーバも含めた7台のサービスサーバを2：1の比率で分配する。各レベルのサーバの台数は次のように決定される。

上位レベル： $7 \times (2/3) = 4.66 \rightarrow 5$ 台

下位レベル： $7 \times (1/3) = 2.33 \rightarrow 2$ 台

中間サーバ2台は下位レベルに含まれるものとするので、下位レベルに割り当てるのは中間サーバ分の台数を除いた

$2 - 2 = 0$ 台

となる。しかし、各レベルに最低1台はサービスサーバが存在しないと不都合が

生じてしまう（例えば、中間サーバが全てレベルアップすると下位レベルのリクエストを処理するサーバが無くなってしまう）。よって、サーバ構成は、次のようになる。

上位：4台 下位：1台 中間サーバ：2台

上記の例では、単純にリクエスト数の比率のみを考慮した例を示したが、サービス毎に処理の重さ（負荷値）が異なる場合、それを考慮する必要がある。

【0056】

例えば、サービスXとサービスYがあり、それぞれの負荷値が

サービスX = 5、サービスY = 10

であるとする。

【0057】

例えば、リクエストのサービス毎の内訳が次のようだとする。

上位レベル：リクエスト総数 = 200（サービスX = 100、サービスY = 100）

下位レベル：リクエスト総数 = 100（サービスX = 20、サービスY = 80）

この場合、リクエスト数の比率は

上位：下位 = 2 : 1

であるが、負荷値の合計はそれぞれ

上位： $100 \times 5 + 100 \times 10 = 1500$

下位： $20 \times 5 + 80 \times 10 = 900$

であり、負荷値の比率は

上位：下位 = 5 : 3

となる。

【0058】

サービスサーバへの負担の量を正確に表しているのはリクエスト数ではなく、負荷値の量であるため、サービスサーバの台数を設定する際も、負荷値の合計比率を使うのが好ましい。

【0059】

そこで、負荷値の合計の比率を用いて計算し直してみると、負荷値合計の比率

は

上位：下位 = 5 : 3

である。この比率に従って 7 台のサービスサーバを分けると

上位： $7 \times (5 / 8) = 4.375 \rightarrow 4$ 台

下位： $7 \times (3 / 8) = 2.625 \rightarrow 3$ 台

中間サーバ分を下位レベルから除くと

上位： 4 台

下位： 1 台

中間サーバ： 2 台

と決定される。

【 0 0 6 0 】

以上がスケジュール設定の基本的な方法であるが、サーバの性能の違いなどを考慮しなくてはならない事項が他にもあるため、実際に運用する際にはもっと細かな計算が必要とされる。しかし、基本的な方法は上記の通りリクエストの比率を元にサーバ台数を割り振ると言うやり方には変わりはない。

【 0 0 6 1 】

また、スケジュール設定の優先順位を考慮することも可能である。すなわち、サービスサーバ構成のスケジュールを立てる際に優先順位 (Priority) を参照する。例えば、優先順位 1 位 (Priority1) が曜日で 2 位 (Priority2) が日にちの場合、曜日別にスケジュールを立て、特に偏りの顕著な日にちのみ別に構成を設定する。

【 0 0 6 2 】

また、スケジュールはサービス管理サーバ内のサーバ情報格納手段で保管され、構成の自動変更時に参照される。

図 1 4 は、本発明の実施形態のシステムブロック図である。

【 0 0 6 3 】

サービス管理サーバは、サービス管理手段 3 0、スケジュール管理手段 3 1、待機管理手段 3 2、サーバ情報格納手段 3 3、及びログ管理手段 3 4 からなっている。

【 0 0 6 4 】

また、サービスサーバグループの各サービスサーバ 3 6 は、負荷計測手段 3 5 を備えている。

サービス管理手段 3 0 は、リクエストを受信して分配先を決定し、サービスサーバのレベルアップ（ダウン）処理を行う。サーバ情報格納手段 3 3 は、システム全体の情報を管理しており、サービス管理サーバで管理されるデータの全てが保管されている。待機管理手段 3 2 は、混雑時に分配不能なリクエストを保管し、サービスサーバの負荷が下がり次第、サービスサーバにリクエストを送信する機能である。スケジュール管理手段 3 1 は、サービスサーバ構成のスケジュールを設定管理する。ログ管理手段 3 4 は、リクエストのログを格納する。また、サービスサーバの負荷計測手段 3 5 は、自分のサービスサーバの負荷状態を監視し、定期的に監視結果をサーバ情報格納手段 3 3 へ送信する。また、サービスサーバで管理するデータを格納している。

【 0 0 6 5 】

受信したリクエストをサービスサーバへ送信する処理を行う場合、1 - 1 に示すように、SLA 情報を含むリクエストを受信し、1 - 2 に示すように、サーバ情報格納手段 3 3 からサーバ情報を取得し、1 - 3 で該当リクエストの分配先を決定する。そして、1 - 4 でリクエストを実行可能な場合、最も負荷が低いサービスサーバ 3 6 にリクエストを送信する。また、1 - 5 のように、リクエストが実行不可能の場合（SLA を維持できない場合）、リクエストを待機管理手段 3 2 に送る。

【 0 0 6 6 】

待機リクエストを送信する処理を行う場合には、2 - 1 で一定間隔でサーバ情報格納手段 3 3 からサーバ情報を取得する。そして、2 - 2 で、実行可能になった時に、サービスサーバ 3 6 にリクエストを送る。

【 0 0 6 7 】

スケジュールを設定する処理を行う場合は、3 - 1 でログ管理手段 3 4 からリクエストのログを取得し、3 - 2 で、スケジュールを設定して、サーバ情報を更新する。

【 0 0 6 8 】

サービスサーバ 3 6 の負荷を計測して通知する処理を行う場合は、4 - 1 に示されるように、サービスサーバ 3 6 の負荷値を計算して、定期的にサーバ情報格納手段 3 3 へ通知する。

【 0 0 6 9 】

図 1 5 は、図 1 4 の各手段が有するデータを示した図である。

サーバ情報格納手段 3 3 は、サーバ識別 I D、閾値、所属グループ、維持する品質レベル、各サービスの負荷値、サーバ性能評価値、サーバ構成変更時間、リクエスト集計優先順、及びサーバ構成スケジュールを有している。

【 0 0 7 0 】

ログ管理手段 3 4 は、リクエスト時間、サービスレベル、要求するサービスなどのデータを有する。

サービスサーバ 3 6 の負荷計測手段 3 5 は、実行中の処理内容、及びサーバ情報格納手段 3 3 が有するデータの内自サーバに関するデータを有する。

【 0 0 7 1 】

図 1 6 は、本発明の実施形態に従ったサービス管理サーバあるいはサービスサーバの機能をプログラムで実現する場合に要求される装置のハードウェア環境を説明する図である。

【 0 0 7 2 】

C P U 4 1 は、バス 4 0 で接続された記憶装置 4 7 あるいは、記録媒体読み取り装置 4 8 を介して可搬記録媒体 4 9 から当該プログラムを読み込み、同じバス 4 0 を介して接続された R A M 4 3 にコピーして実行する。C P U 4 1 にバス 4 0 を介して接続される R O M 4 2 には、B I O S などの基本プログラムが格納されるが、本発明の実施形態を実現するプログラムを格納してもよい。

【 0 0 7 3 】

入出力装置 5 0 は、バス 4 0 を介して C P U 4 1 に接続され、C P U 4 1 の演算結果を装置のユーザに提示したり、ユーザからの指示を C P U 4 1 に伝えるために使用され、例えば、キーボード、マウス、タブレット、ディスプレイなどからなる。

【 0 0 7 4 】

通信インターフェース 4 4 は、ネットワーク 4 5 を介して図 1 6 の装置が情報提供者 4 6 と通信するために使用される。本発明の実施形態を実現するプログラムを情報提供者 4 6 からダウンロードし、CPU 4 1 が実行しても良いし、ネットワーク環境下で、当該プログラムを実行することも可能である。また、通信インターフェース 4 4 を介して、サービス管理サーバとサービスサーバとが通信したり、ウェブサーバと通信することも可能である。

【 0 0 7 5 】

(付記 1) 情報装置に、ネットワークを介してクライアントからのサービス要求に応じたサービスを提供するサービスサーバを複数收容し、該複数のサービスサーバにサービス要求を配分するサービス管理方法を実現させるプログラムにおいて、

該複数のサービスサーバを、提供するサービスの品質レベル毎の複数のグループのサービスサーバと、該グループ間を移行して、移行先のグループのサービス品質でサービスを提供する中間サーバグループのサービスサーバとにグループ化して管理する管理ステップと、

いずれかのグループのサービスサーバの負荷が増加し、そのグループが提供すべき品質レベルを維持できなくなる場合に、該中間サーバグループの最も負荷が低いサービスサーバを少なくとも 1 つ、該グループのサービスサーバとして使用して、該グループのサービスサーバの負荷の低減を図る中間サーバ移行ステップと、

を備えることを特徴とするサービス管理方法を情報装置に実現させるプログラム

。

【 0 0 7 6 】

(付記 2) 前記管理ステップは、

前記グループ化されたサービスサーバが、どのグループに属するかの情報を格納する格納手段を更に備えることを特徴とする付記 1 に記載のプログラム。

【 0 0 7 7 】

(付記 3) 前記サービスの品質は、前記サービスサーバの応答時間であるこ

とを特徴とする付記 1 に記載のプログラム。

(付記 4) 前記サービス要求の履歴を記録するログ管理ステップと、

該ログ管理手段の記録に基づいて、日にちあるいは曜日毎にスケジュールを作成し、作成したスケジュールに従って前記グループ分けの仕方を変更するスケジュール管理ステップと、

を更に備えることを特徴とする付記 1 に記載のプログラム。

【 0 0 7 8 】

(付記 5) 前記各サービスサーバは、自サーバがサービス要求を処理するために必要とする負荷値を計測する負荷計測ステップを有し、

該負荷計測ステップから報告される各サービスサーバの負荷値に基づいて、前記中間サーバグループのサービスサーバを別のグループに移行させることを特徴とする付記 1 に記載のプログラム。

【 0 0 7 9 】

(付記 6) ネットワークを介してクライアントからのサービス要求に応じたサービスを提供するサービスサーバを複数收容し、該複数のサービスサーバにサービス要求を配分するサービス管理方法において、

該複数のサービスサーバを、提供するサービスの品質レベル毎の複数のグループのサービスサーバと、該グループ間を移行して、移行先のグループのサービス品質でサービスを提供する中間サーバグループのサービスサーバとにグループ化して管理する管理ステップと、

いずれかのグループのサービスサーバの負荷が増加し、そのグループが提供すべき品質レベルを維持できなくなる場合に、該中間サーバグループのサービスサーバを少なくとも 1 つ、該グループのサービスサーバとして使用して、該グループのサービスサーバの負荷の低減を図る中間サーバ移行ステップと、

を備えることを特徴とするサービス管理方法。

【 0 0 8 0 】

(付記 7) ネットワークを介してクライアントからのサービス要求に応じたサービスを提供するサービスサーバを複数收容し、該複数のサービスサーバにサービス要求を配分するサービス管理方法を情報装置に実行させるプログラムにお

いて、該サービス管理方法は、

該複数のサービスサーバを、提供するサービスの品質レベル毎の複数のグループのサービスサーバと、該グループ間を移行して、移行先のグループのサービス品質でサービスを提供する中間サーバグループのサービスサーバとにグループ化して管理する管理ステップと、

いずれかのグループのサービスサーバの負荷が増加し、そのグループが提供すべき品質レベルを維持できなくなる場合に、該中間サーバグループのサービスサーバを少なくとも1つ、該グループのサービスサーバとして使用して、該グループのサービスサーバの負荷の低減を図る中間サーバ移行ステップと、
を備えることを特徴とするプログラム。

【 0 0 8 1 】

（付記8）ネットワークを介してクライアントからのサービス要求に応じたサービスを提供するサービスサーバを複数收容し、該複数のサービスサーバにサービス要求を配分するサービス管理方法を情報装置に実行させるプログラムを格納した、情報装置読み取り可能な記録媒体において、該サービス管理方法は、

該複数のサービスサーバを、提供するサービスの品質レベル毎の複数のグループのサービスサーバと、該グループ間を移行して、移行先のグループのサービス品質でサービスを提供する中間サーバグループのサービスサーバとにグループ化して管理する管理ステップと、

いずれかのグループのサービスサーバの負荷が増加し、そのグループが提供すべき品質レベルを維持できなくなる場合に、該中間サーバグループのサービスサーバを少なくとも1つ、該グループのサービスサーバとして使用して、該グループのサービスサーバの負荷の低減を図る中間サーバ移行ステップと、
を備えることを特徴とする記録媒体。

【 0 0 8 2 】

（付記9）ネットワークを介してクライアントからのサービス要求に応じたサービスを提供するサービスサーバを複数收容し、該複数のサービスサーバにサービス要求を配分するサービス管理装置において、

該複数のサービスサーバを、提供するサービスの品質レベル毎の複数のグルー

プのサービスサーバと、該グループ間を移行して、移行先のグループのサービス品質でサービスを提供する中間サーバグループのサービスサーバとにグループ化して管理する管理手段と、

いずれかのグループのサービスサーバの負荷が増加し、そのグループが提供すべき品質レベルを維持できなくなる場合に、該中間サーバグループの最も負荷が低いサービスサーバを少なくとも1つ、該グループのサービスサーバとして使用して、該グループのサービスサーバの負荷の低減を図る中間サーバ移行手段と、を備えることを特徴とするサービス管理装置。

【 0 0 8 3 】

【発明の効果】

本発明によれば、サービスを行うサービスサーバ間の負荷の差を適切に均等化しつつ、サービスの品質を維持したサービスの提供を行うためのサービス管理装置を提供することが出来る。

【図面の簡単な説明】

【図 1】

本発明の実施形態の概略を示す図である。

【図 2】

本発明の実施形態が適用されるシステムの構成図である。

【図 3】

中間グループのサービスサーバ（中間サーバ）の状態変化を示す図である。

【図 4】

サービス管理サーバの処理の概略を示す図である。

【図 5】

サービス管理サーバの処理の詳細を示すフローチャートである。

【図 6】

図 5 のステップ S 1 3、S 1 4 の詳細を示すフローチャートである。

【図 7】

図 5 のステップ S 1 9 を詳細に示したフローチャートである。

【図 8】

サービス負荷の自動計測処理の流れを示す図である。

【図 9】

サービス負荷の自動計測処理を具体的に示すフローチャートである。

【図 1 0】

サービスサーバの運用スケジュールを設定する実施形態の概念を示す図である。

【図 1 1】

スケジュールの設定処理の概略フローを示す図である。

【図 1 2】

スケジュール設定処理のより具体的なフローチャートである。

【図 1 3】

スケジュールに基づいてサービスサーバの構成を変更する際の処理を示すフローチャートである。

【図 1 4】

本発明の実施形態のシステムブロック図である。

【図 1 5】

図 1 4 の各手段が有するデータを示した図である。

【図 1 6】

本発明の実施形態に従ったサービス管理サーバあるいはサービスサーバの機能をプログラムで実現する場合に要求される装置のハードウェア環境を説明する図である。

【図 1 7】

A S P サービスを提供するシステムの概略構成図である。

【図 1 8】

S L A におけるサービス管理サーバのサービス管理方法の従来技術を説明する図である。

【図 1 9】

従来の問題点を説明する図である。

【符号の説明】

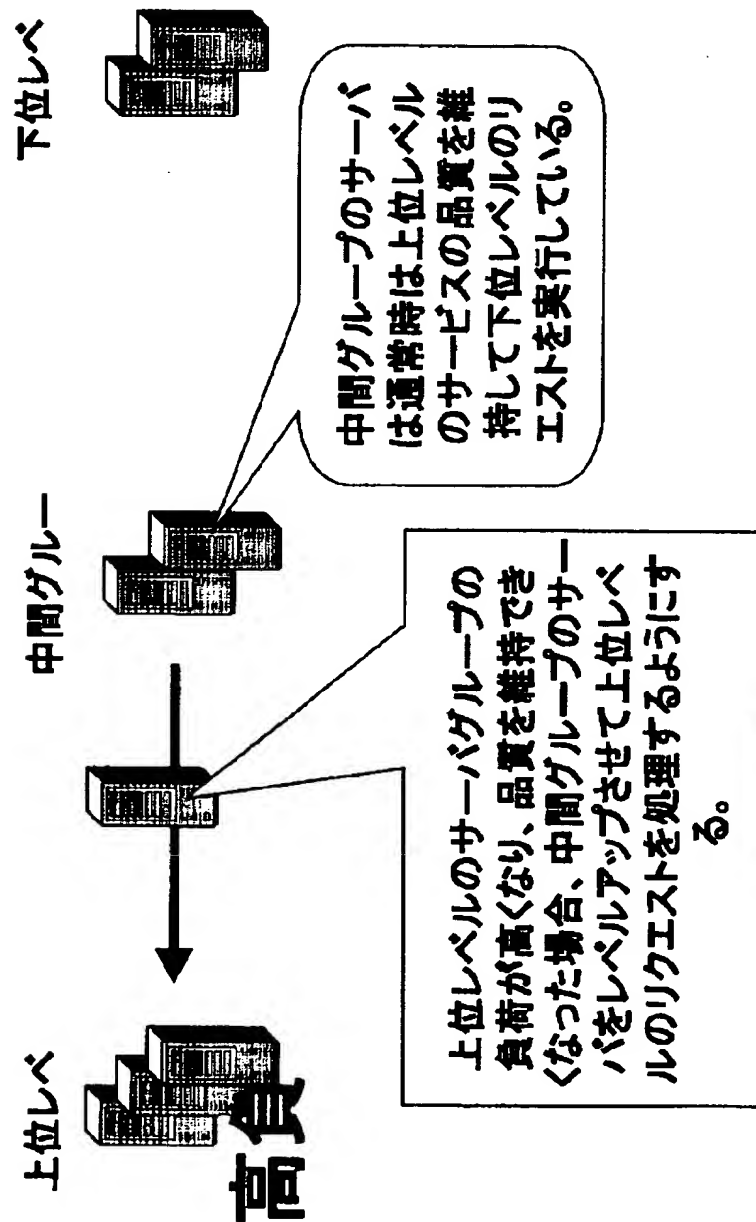
- 2 0 インターネット
- 2 1 ファイアウォール
- 2 2 ウェブサーバ
- 2 3 サーブレットエンジン
- 2 4 サービス管理サーバ
- 2 5 - 1 ~ 2 5 - n サービスサーバ
- 2 6 データベースサーバ
- 3 0 サービス管理手段
- 3 1 スケジュール管理手段
- 3 2 待機管理手段
- 3 3 サーバ情報格納手段
- 3 4 ログ管理手段
- 3 5 負荷計測手段
- 3 6 サービスサーバ

【書類名】

図面

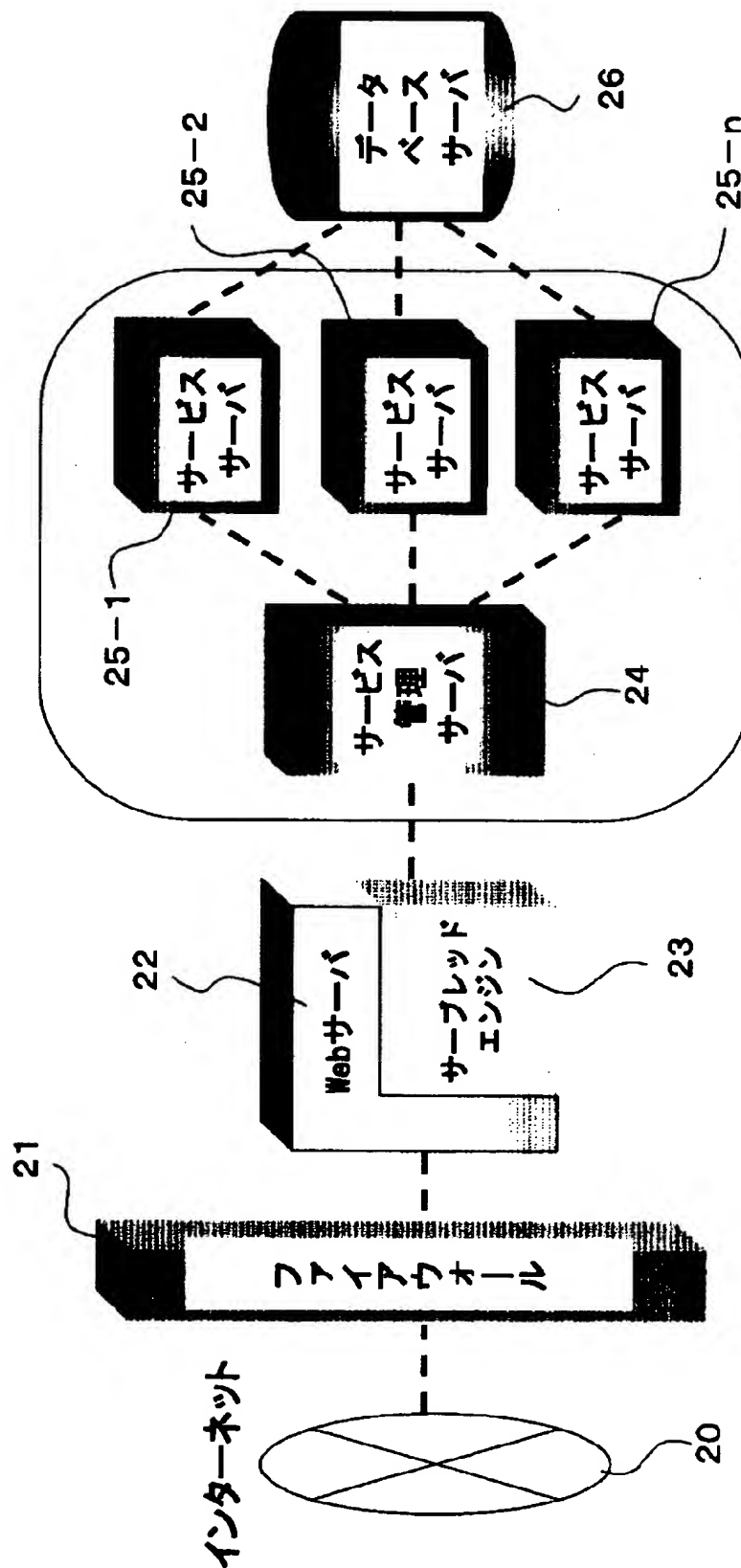
【図 1】

本発明の実施形態の概略を示す図



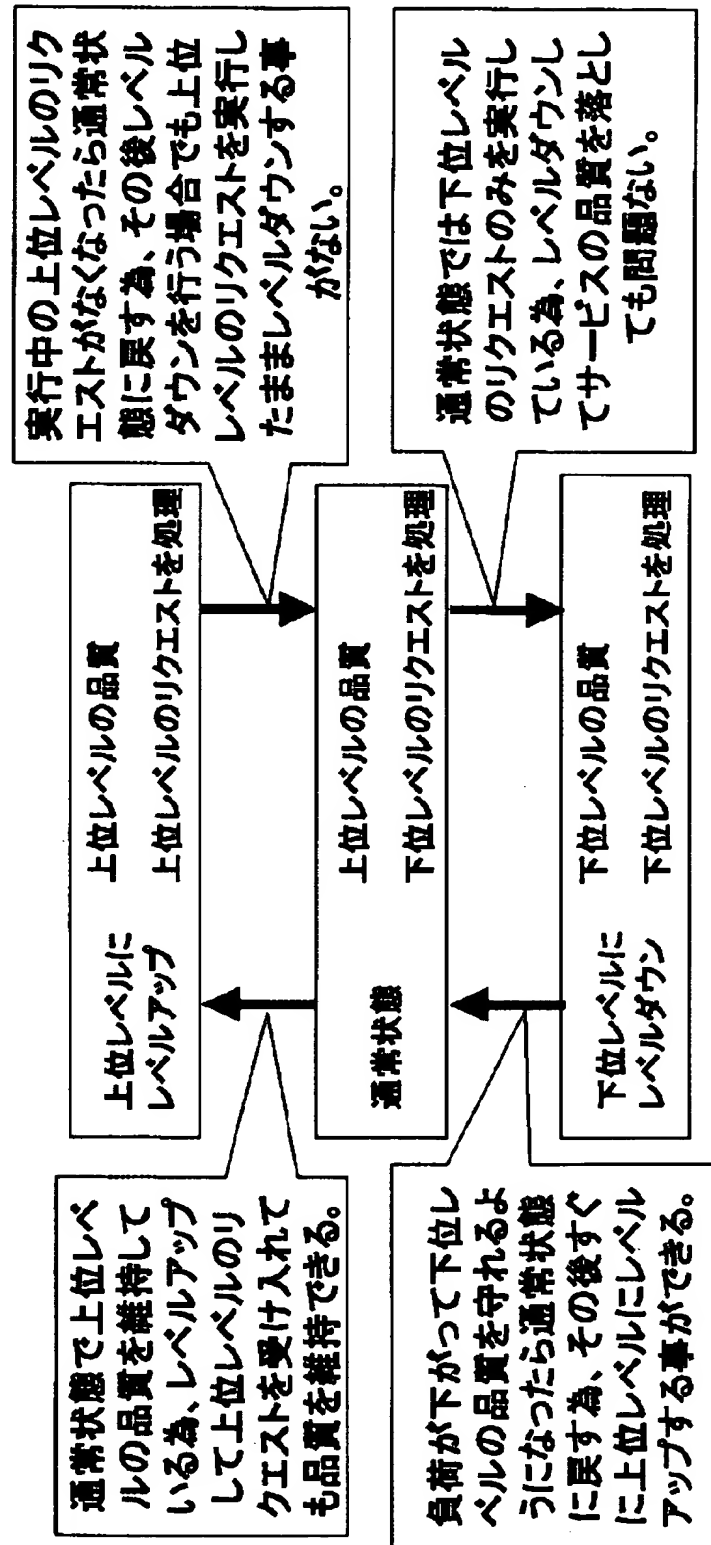
【図 2】

本発明の実施形態が適用されるシステムの構成図



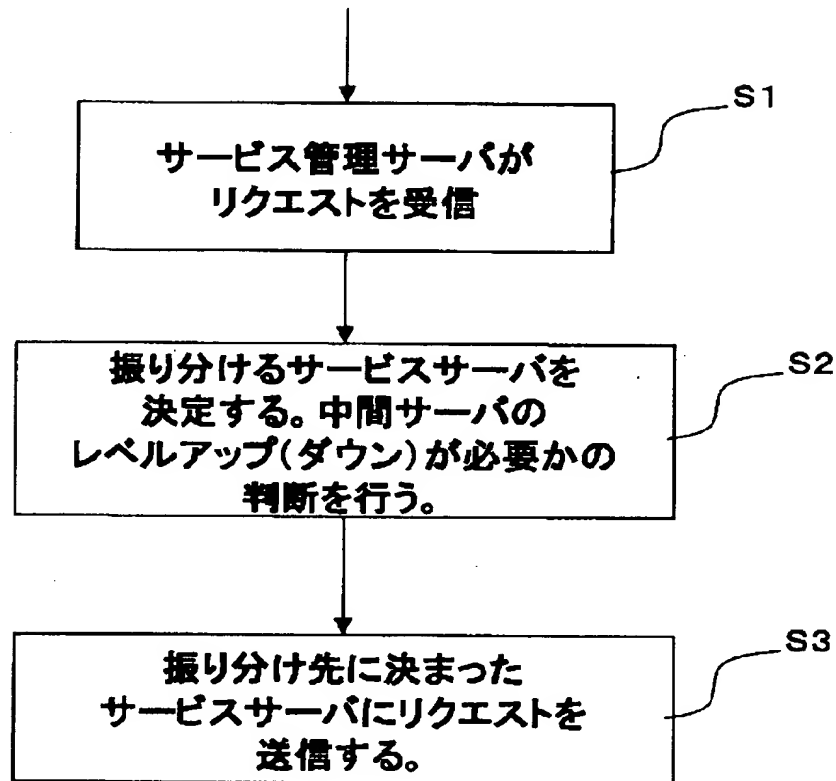
【図 3】

中間グループのサービスサーバ(中間サーバ)の状態変化を示す図



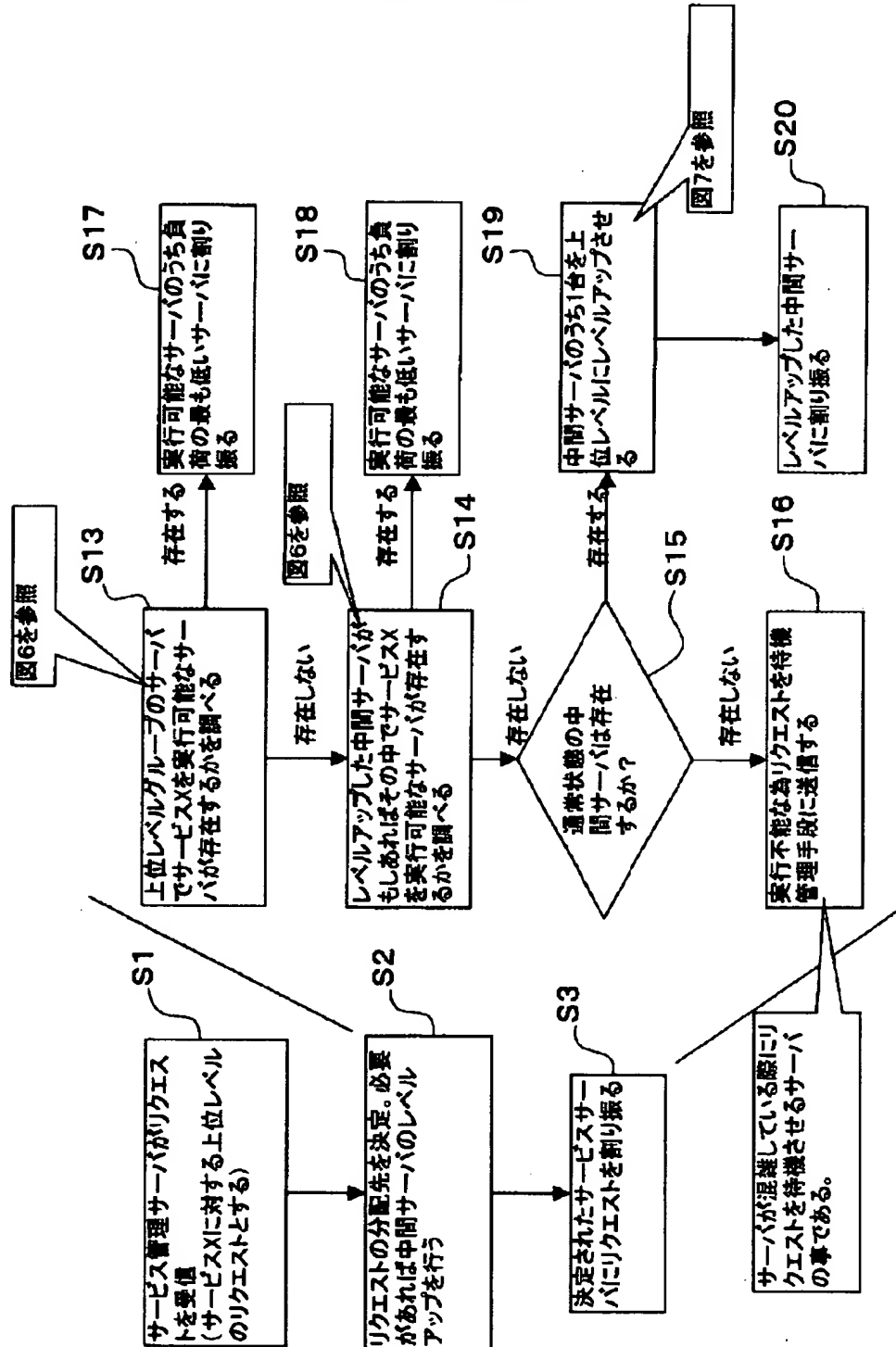
【図 4】

サービス管理サーバの処理の概略を示す図



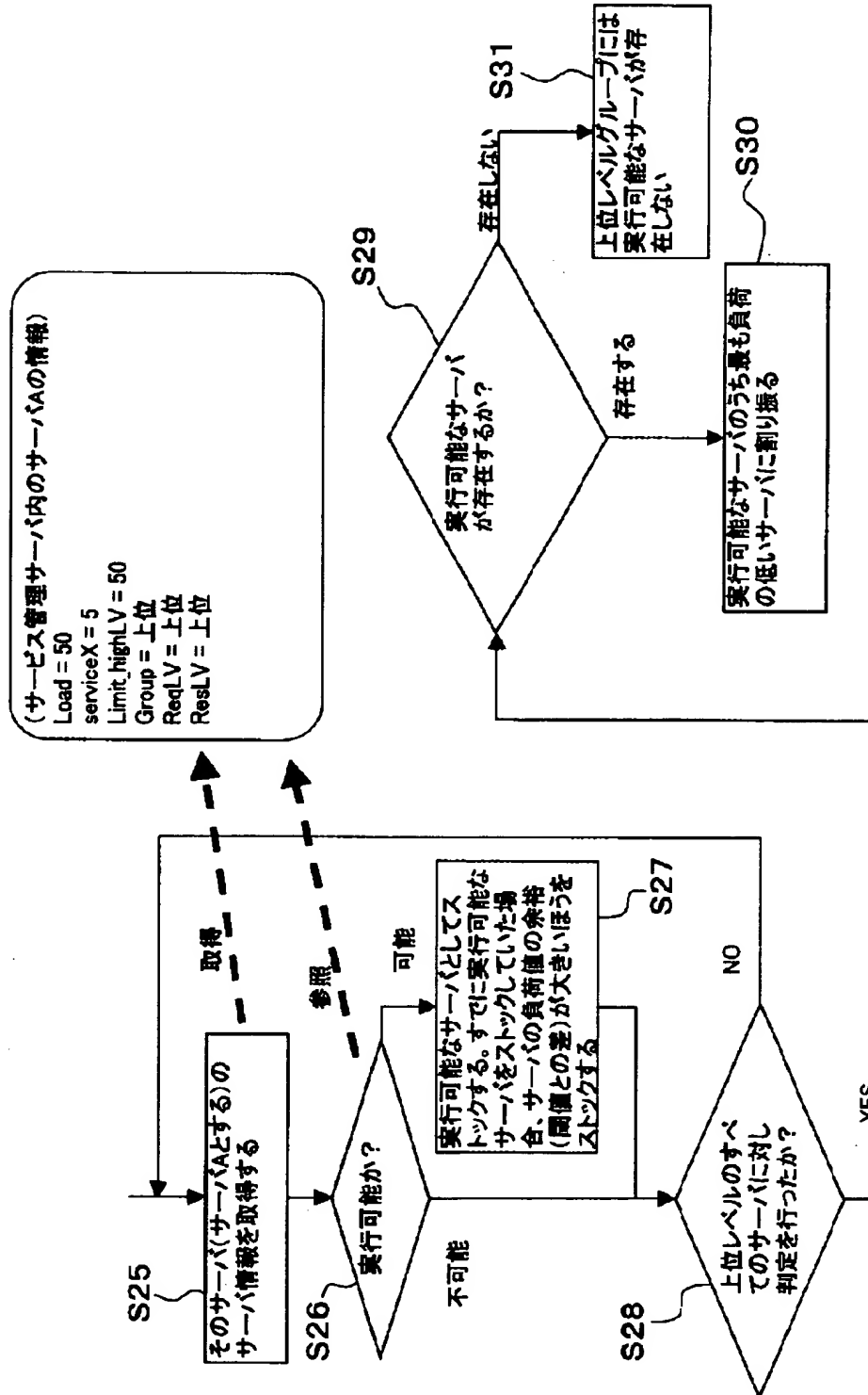
【図 5】

サービス管理サーバの処理の詳細を示すフローチャート



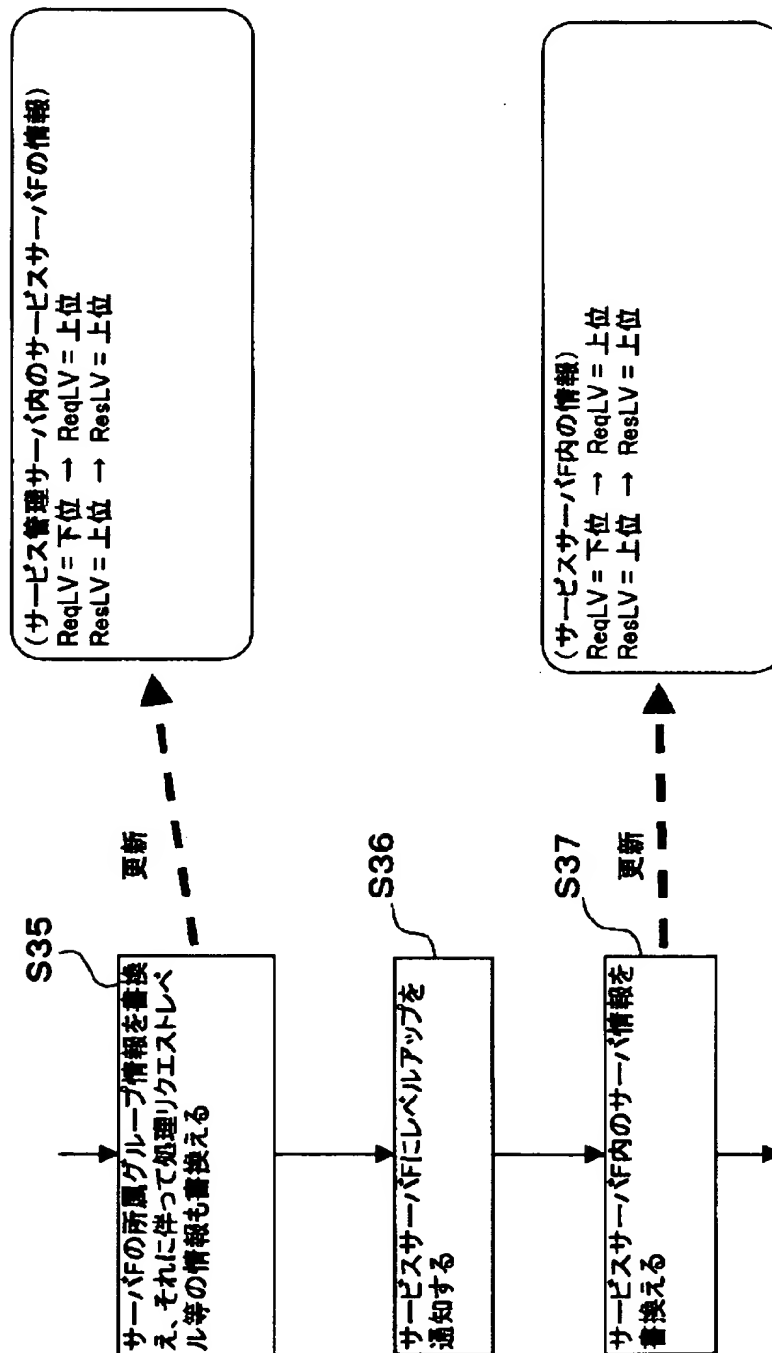
【図 6】

図5のステップS13、S14の詳細を示すフローチャート



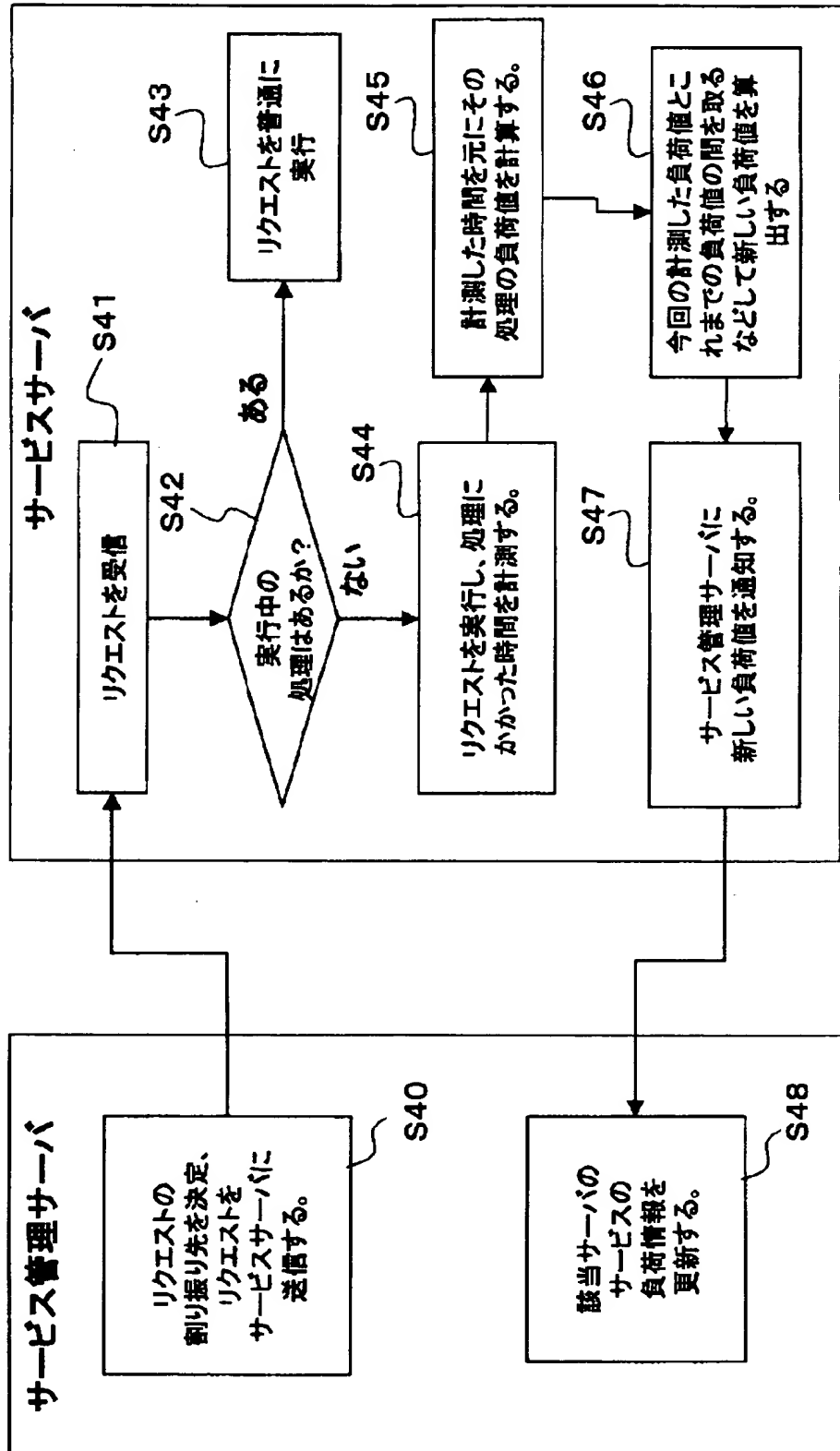
【図 7】

図5のステップS19を詳細に示したフローチャート



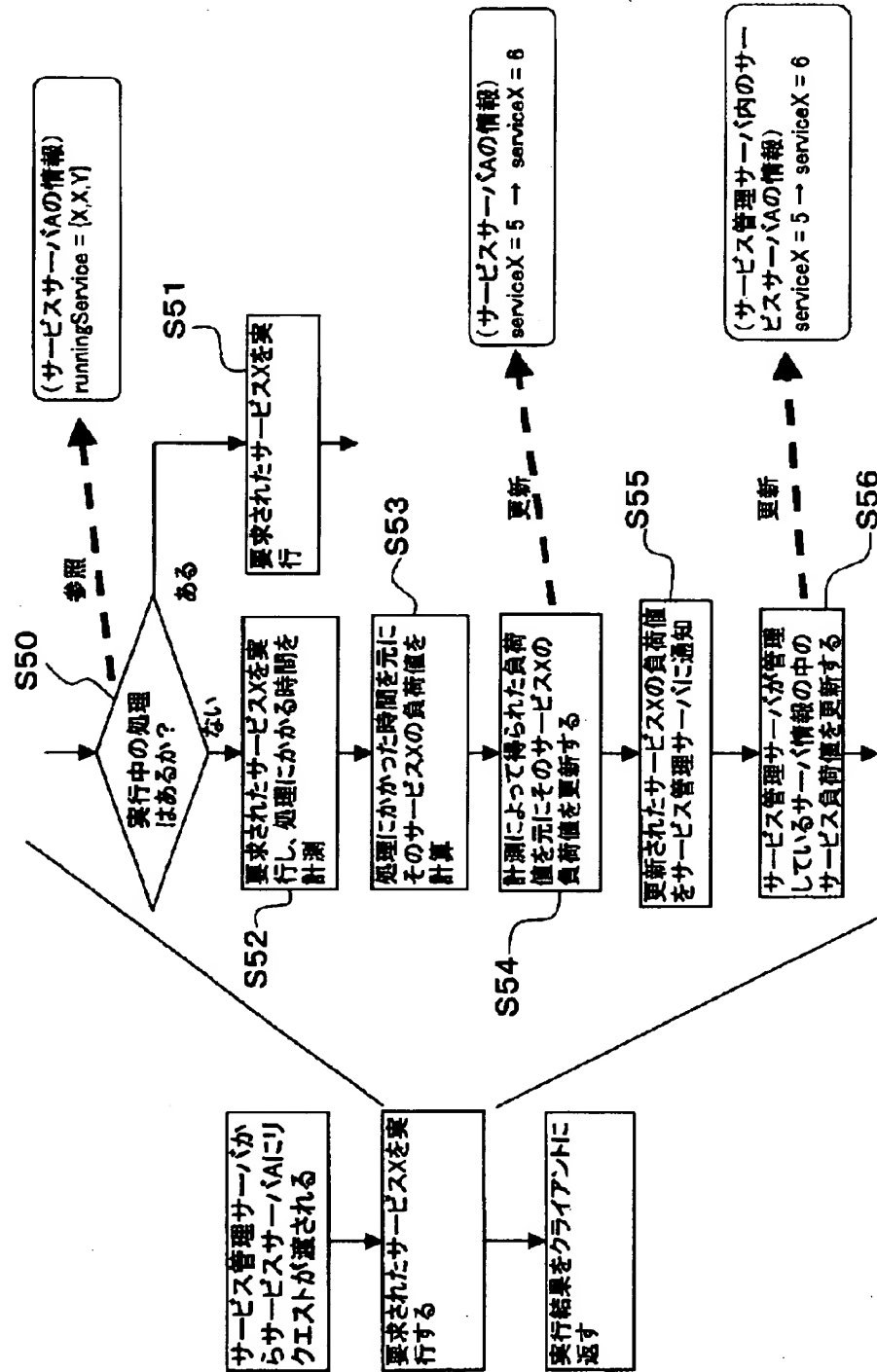
【図 8】

サービス負荷の自動計測処理の流れを示す図



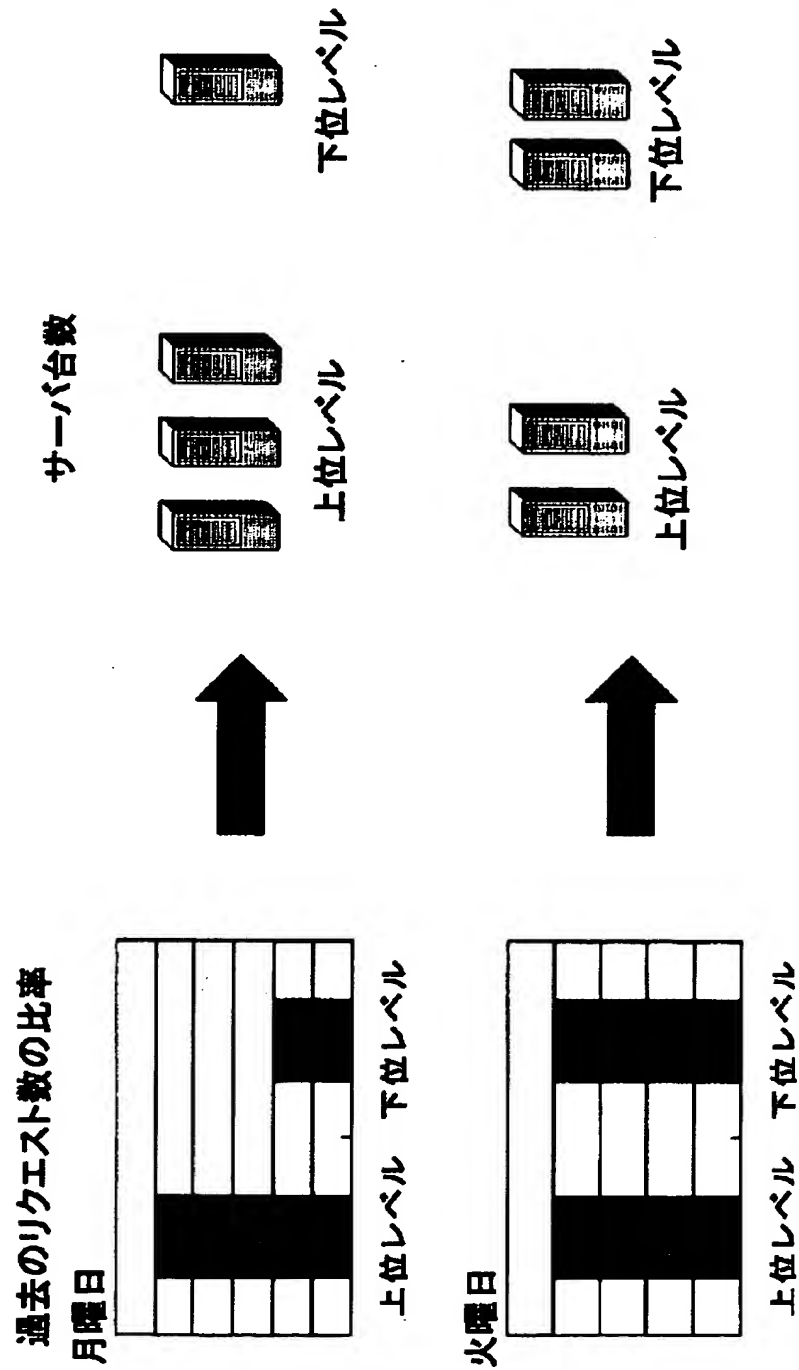
【図 9】

サービス負荷の自動計測処理を具体的に示すフローチャート



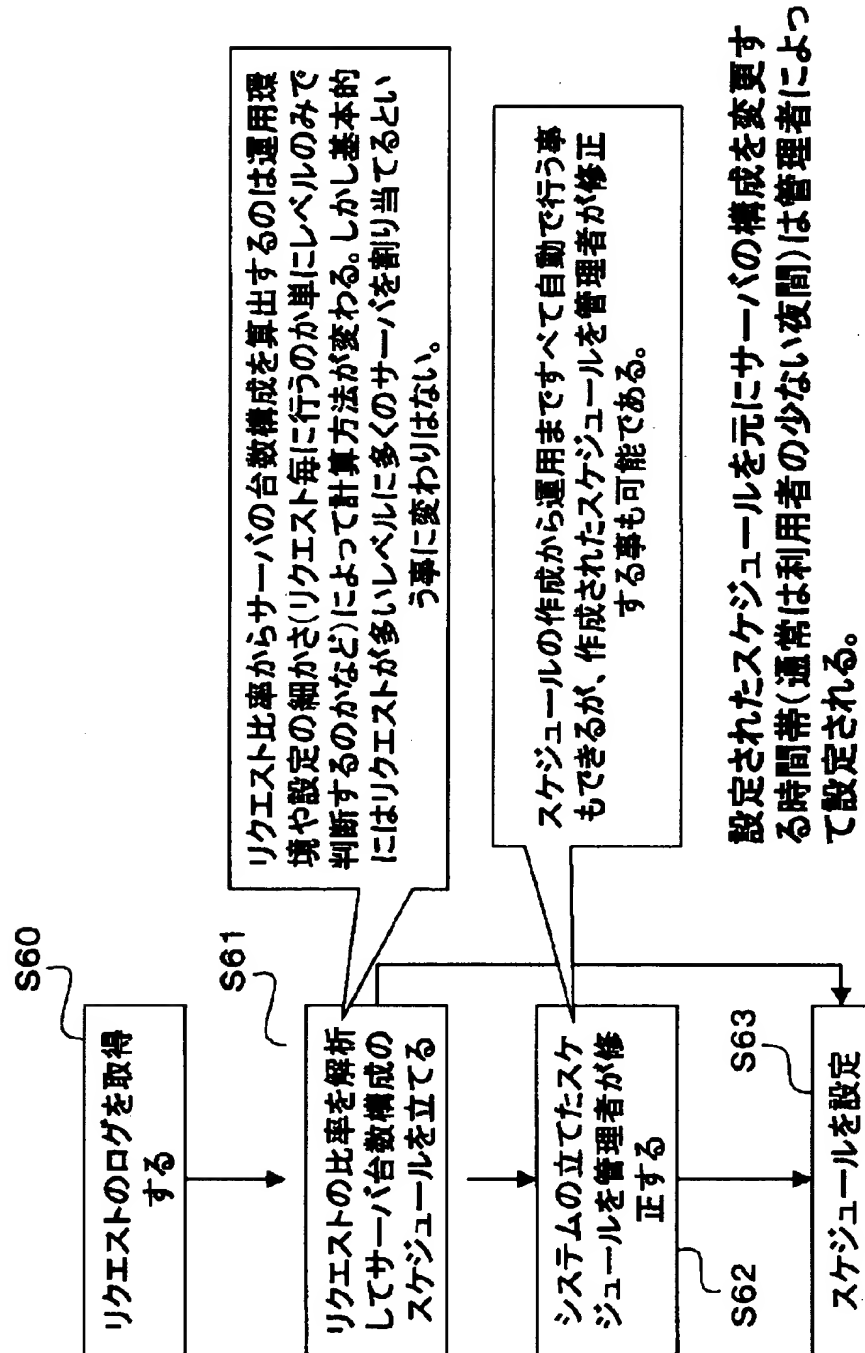
【図 1 0】

サービスサーバの運用スケジュールを設定する実施形態の概念を示す図



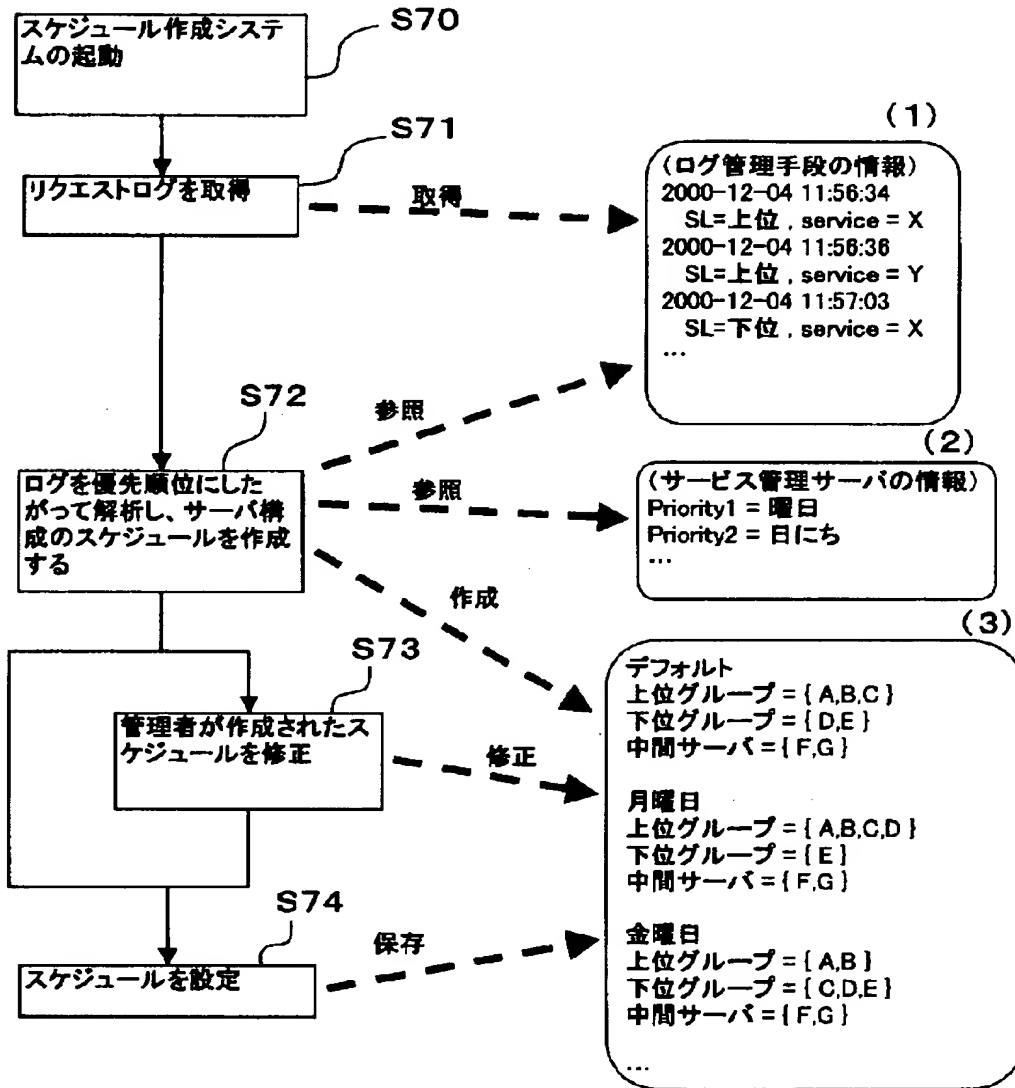
【図 1 1】

スケジュールの設定処理の概略フローを示す図



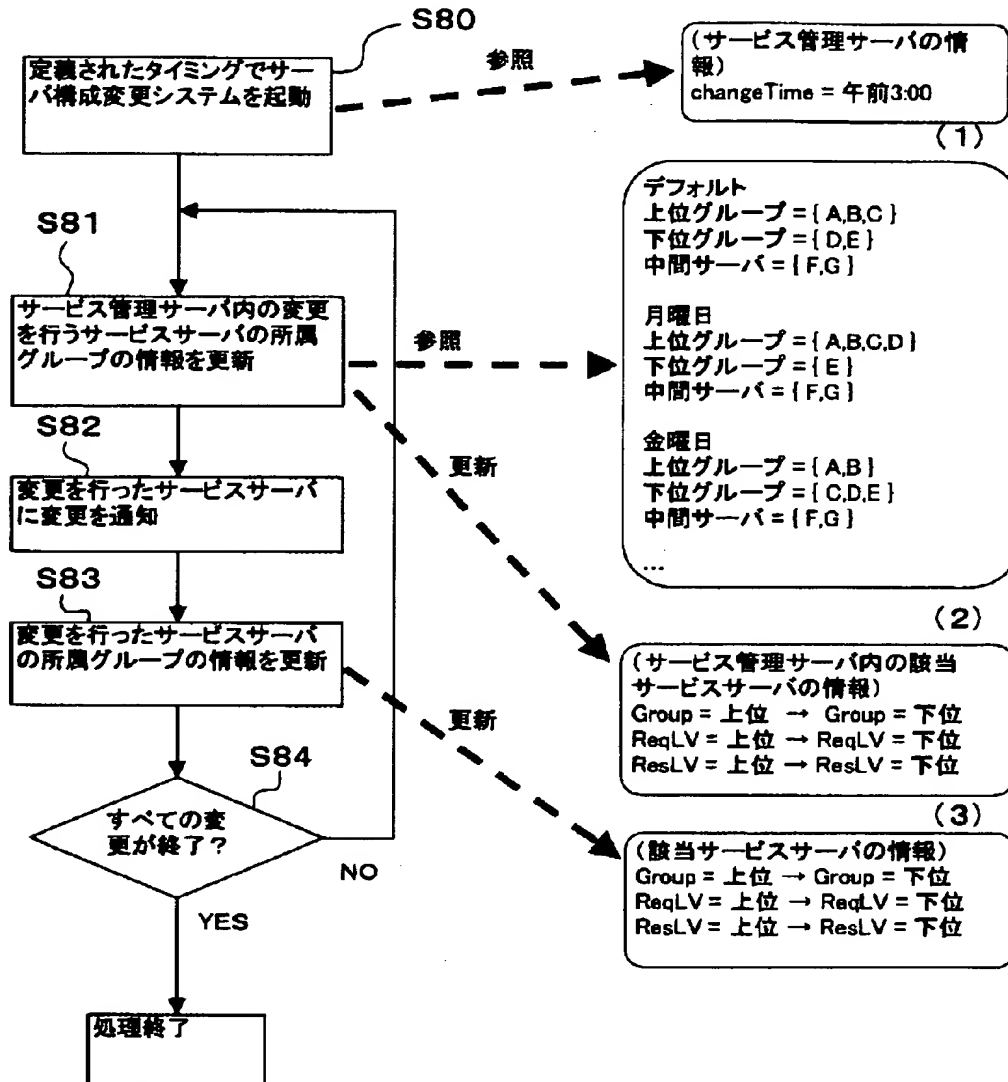
【図 12】

スケジュール設定処理のより具体的なフローチャート



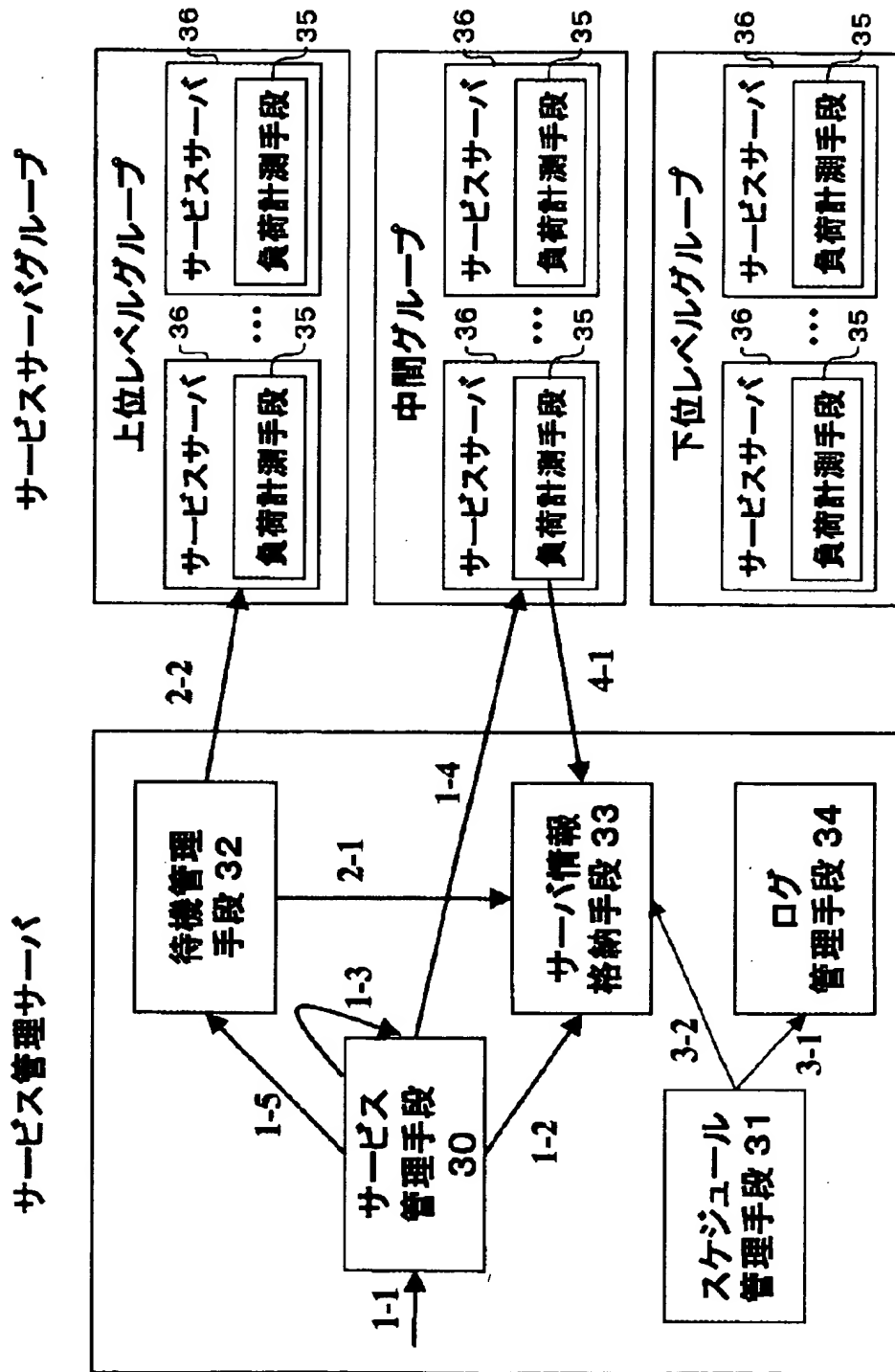
【図13】

スケジュールに基づいてサービスサーバの
構成を変更する際の処理を示すフローチャート



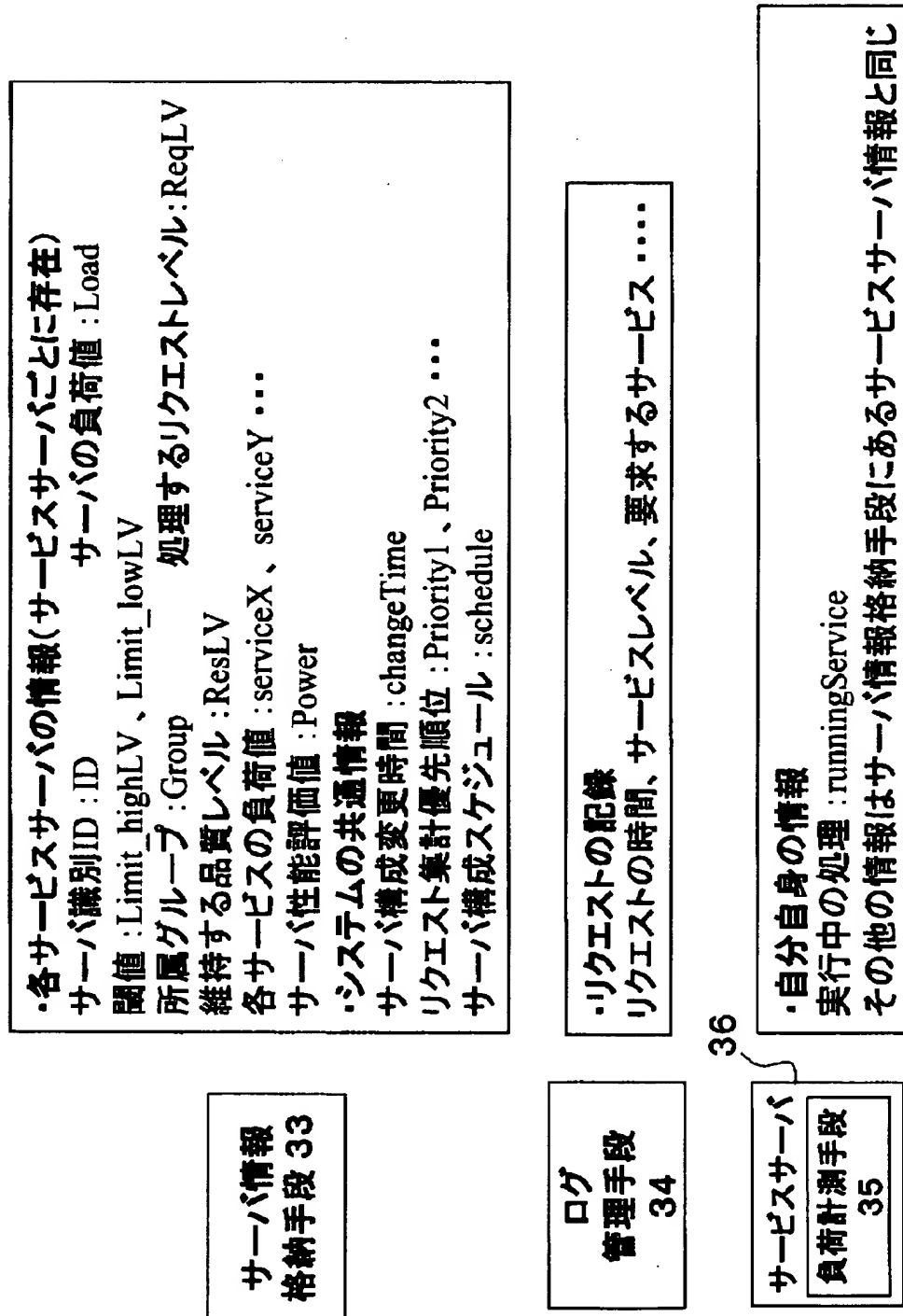
【図 14】

本発明の実施形態のシステムブロック図



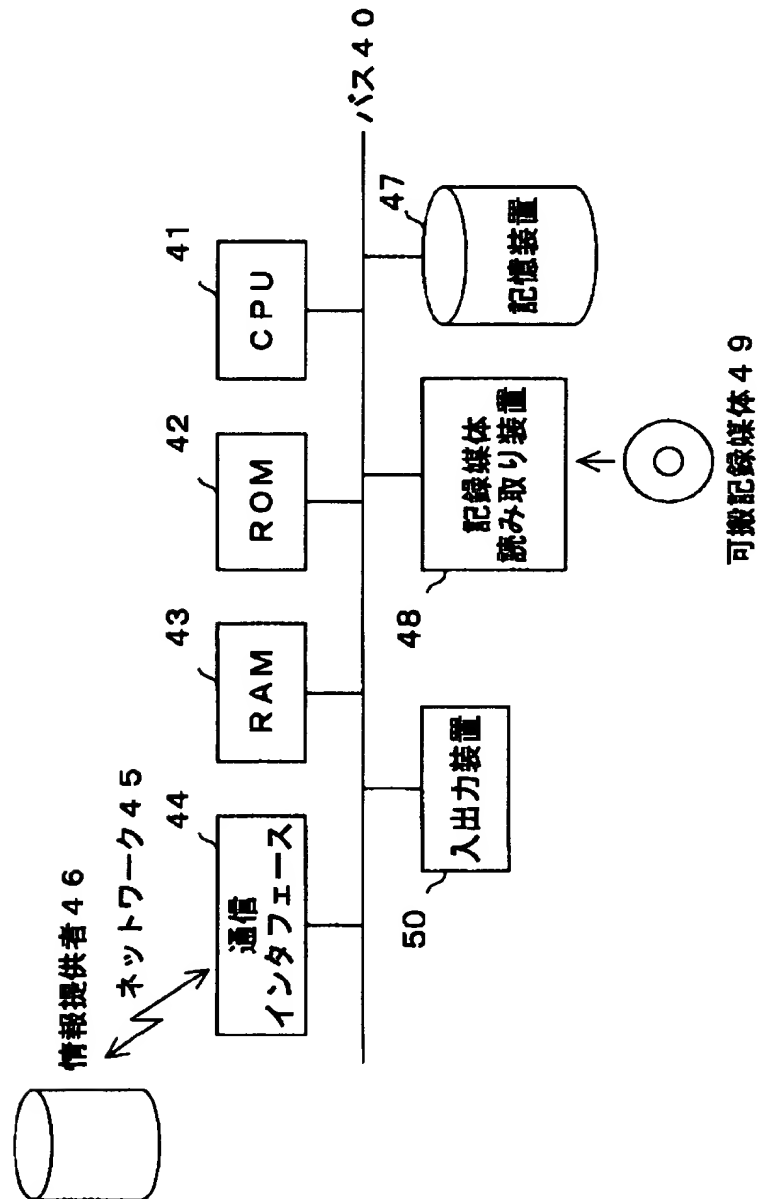
【図 15】

図 14 の各手段が有するデータを示した図



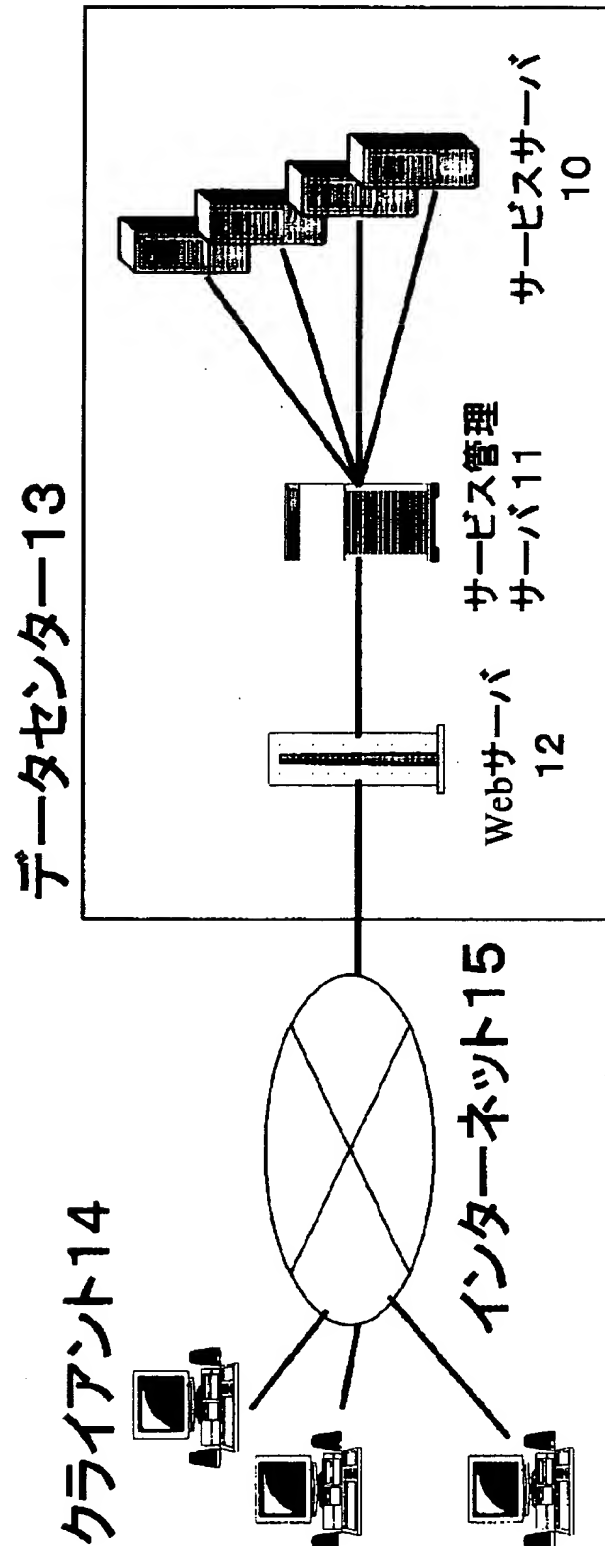
【図 16】

本発明の実施形態に従ったサービス管理サーバあるいはサービスサーバの機能をプログラムで実現する場合に要求される装置のハードウェア環境を説明する図



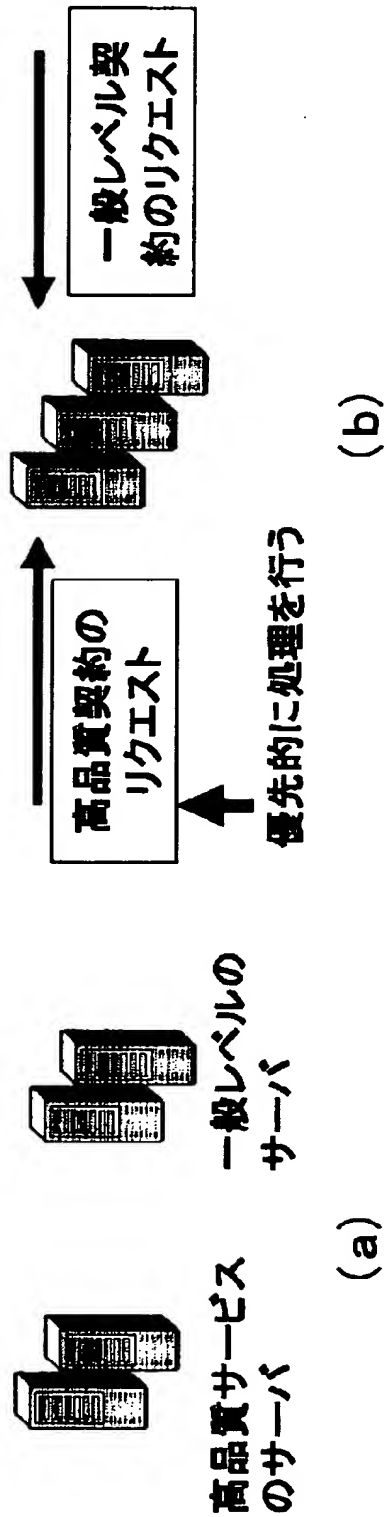
【図 17】

ASPサービスを提供するシステムの概略構成図



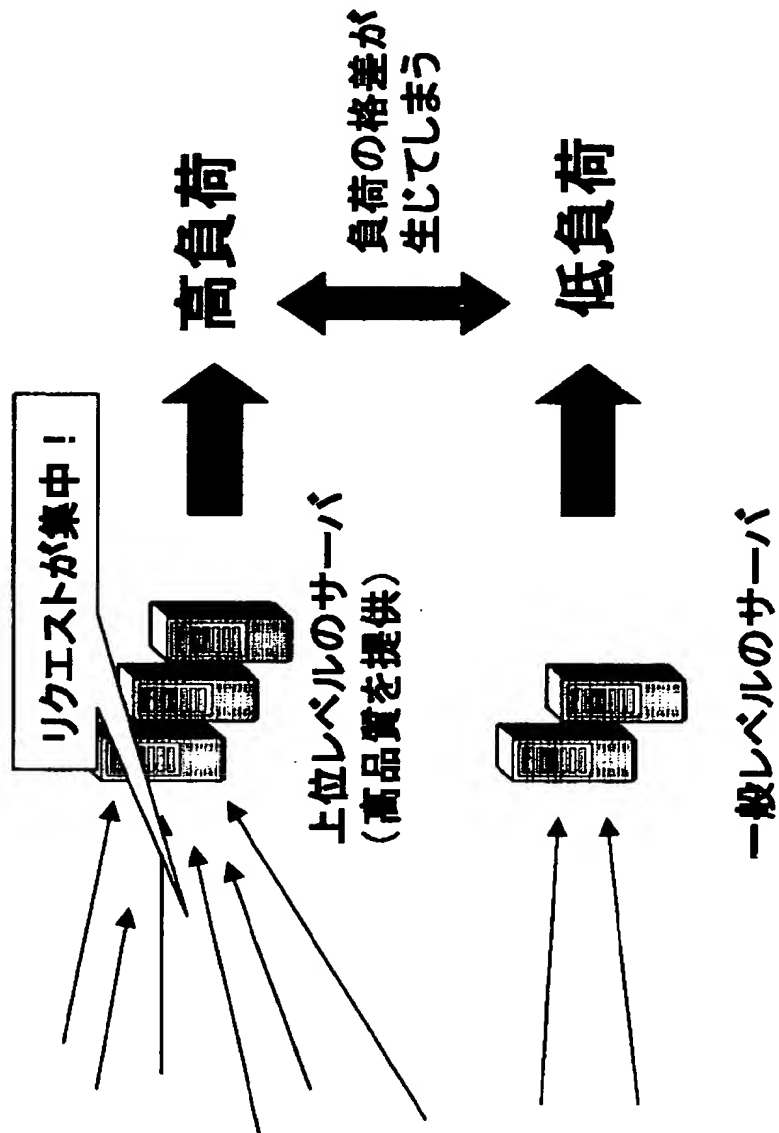
【図 1 8】

SLAにおけるサービス管理サーバの
サービス管理方法の従来技術を説明する図



【図19】

従来の問題点を説明する図



【書類名】 要約書

【要約】

【課題】 サービスの品質を維持しつつ、サーバの負荷を適切に分散することの出来るサービス管理装置を提供する。

【解決手段】 インターネットに接続されたウェブサーバを介してクライアントからサービス要求を受け付け、サービスを提供するサービスサーバが複数台、サービス要求を振り分けるサービス管理サーバに接続された構成を有している。サービスサーバは、提供するサービスの品質に応じて複数のレベルにグループ化される。更に、これらのレベルの他に中間サーバと呼ばれるサービス提供品質レベルを可変とするサービスサーバを設ける。サービス管理サーバは、何れかのレベルのグループのサービスサーバの負荷が大きくなると、中間サーバをそのグループのサーバとして使用することによって、そのグループの負荷を軽減する。これにより、負荷をグループ間で均一にしつつ、品質の維持を図る。

【選択図】 図 1

出 願 人 履 歴 情 報

識別番号 [000005223]

1. 変更年月日	1996年 3月26日
[変更理由]	住所変更
住 所	神奈川県川崎市中原区上小田中4丁目1番1号
氏 名	富士通株式会社